Enregistreur autonome de décharge de Batteries

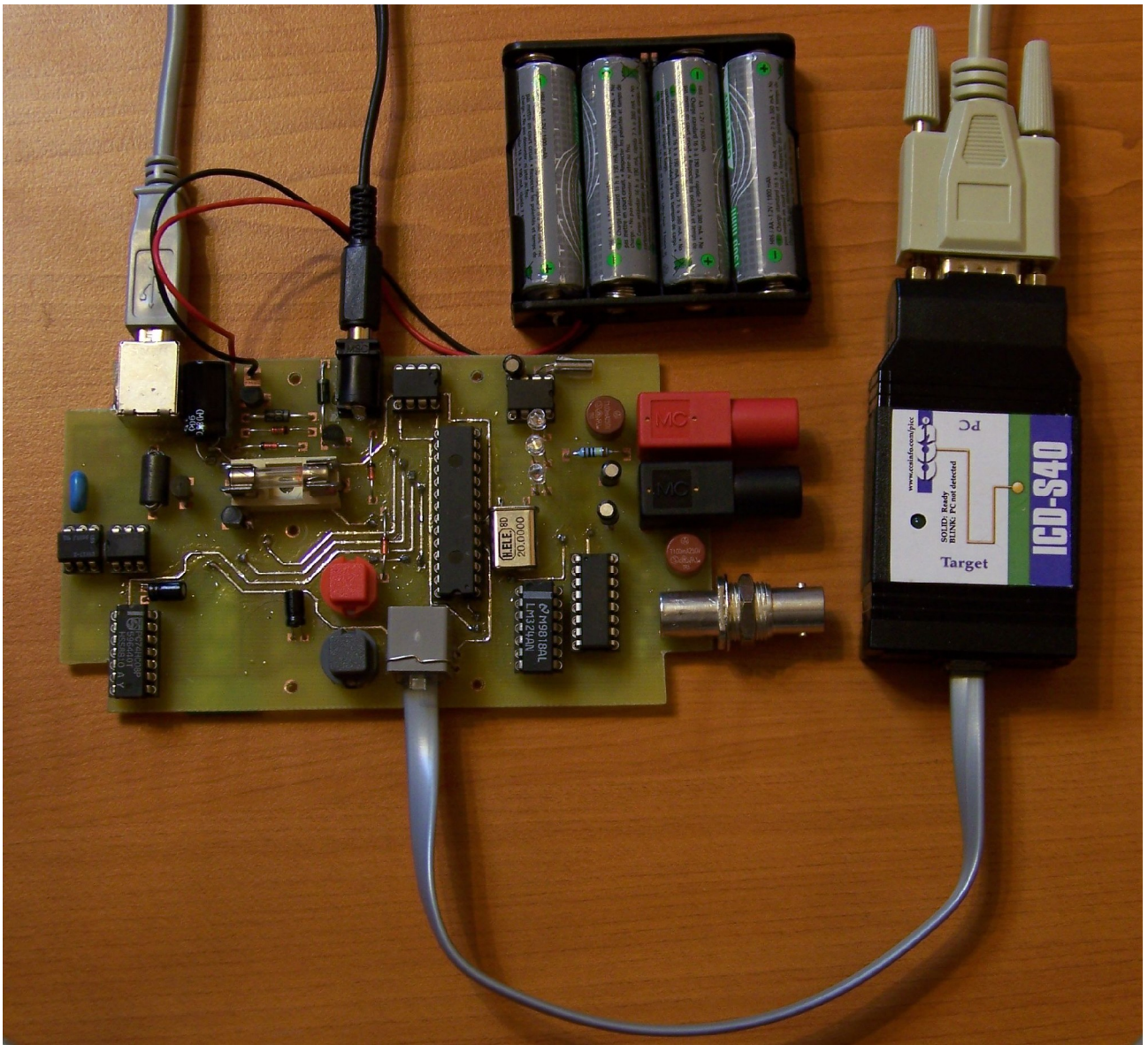


Table des matières

Cahier des charges :....3
Société intéressée :....3
Schéma bloc simplifié :....3
Données techniques :....3
Dessin du boîtier:....4
Mode d'emploi :....4
Par liaison Bluetooth ou USB on peut observer l'appareil :....5
Schéma fonctionnel :....6
FS1 : Interfaces Tension et Courant :....6
FS2 : Mémoire et horloge temps réel :....6
FS3 : Commande et signalisation :....6
FS4 : Interface USB :....6
FS5 : Interface Bluetooth :....6
FS6 : Alimentations :....6
FS7 : Fonctions logicielles :....6
Schéma structurel :....7
Alimentations :....8
Schéma structurel :....8
Bouton rouge « ON » :....8
Tension d'alimentation :....8
Surveillance de la charge et décharge des 4 éléments NiMh :....9
Limitation de tension :....9
Charge d'entretien :....9
Anti retour :....9
Interfaces tension et courant :....10
Schéma structurel :....10
Étude commune aux 2circuits :....10
Mesure du courant :....11
Mesure de la tension:....12
Potentiomètre numérique DS1803 :....12
Commande et signalisation :....13
Mémoire et horloge temps réel :....13
EEPROM externe 24LC256 :....13
Horloge temps réel PCF8583 :....13
Interface USB :....14
Schéma structurel :....14
Brochages des connecteurs :....14
Calculs autour du photocoupleur :....15
Interface Bluetooth :....16
Schéma structurel :....16
Schéma de raccordement de la carte sur un ordinateur :....16
Procédure de raccordement de la carte sur l'interface ordinateur :....16
Free2move, configuration du module :....17
Test de la liaison :....17
Fonctions logicielles :....19
Reprogrammation du PIC16F876A :....20
Circuit imprimé :....20
Bon de commande :....22
Planning d'étude et de réalisation :....23
Mesures de validation :....24
Fonctions logicielles :....25
Organisation en mémoire EEPROM Externe :....25
Organisation en mémoire EEPROM Interne :....25
Algorithmes :....25
Programme langage C :....28
Annexes :....35

Enregistreur autonome de décharge de batteries

Cahier des charges :

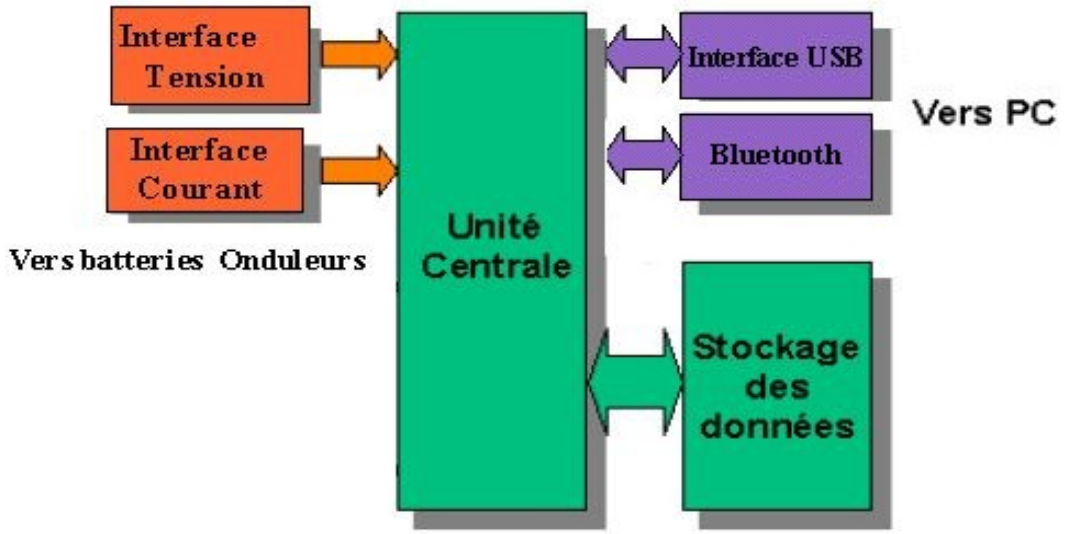
Société intéressée :

La société BSAV sise à Vénissieux fait la maintenance d'Alimentations Sans Interruption (ASI) dédiées à des systèmes informatiques. Elles sont composées d'un redresseur, d'un élévateur de tension, d'un onduleur et d'un stockage d'énergie sur batterie. Il faut contrôler régulièrement le vieillissement de la batterie pour que sa réserve reste importante.

La société BSAV contrôle les « ASI » de ses clients tous les 6 mois.

L'appareil à réaliser doit relever le courant et la tension d'une batterie en décharge sur un onduleur. La finalité est de tracer 2 courbes graphiques U=f(t) et I=f(t) pour les comparer à celles d'origine et d'en déduire si la batterie est toujours utilisable.

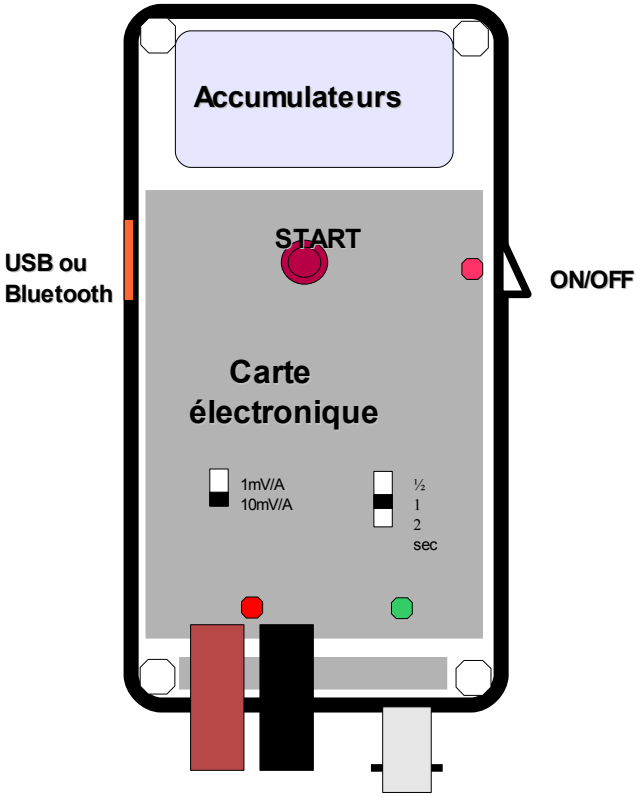
Schéma bloc simplifié :



Données techniques :

- Plage de tension : -400 à 400V.
- Plage de courant : -100 à 100 A..
- Plage de mesure : 30 minutes minimum.
- Périodes d'échantillonnage : 1s à 4min.
- Coffret équipé de deux voies d'entrées :
  - Voie 1 : mesure de la tension continue à l'aide de 2 pointes de touche sur douille de sécurité.
  - Voie 2 : Pince de courant sur connexion BNC.
- Le boîtier est équipé de Leds indiquant la présence du courant ou de la tension pour chaque voie et qui servent aussi à d'autres usages. (Voir mode d'emploi.)
- Alimentation du module par accumulateurs LR6.

Dessin du boîtier:
(Non contractuel)



Mode d'emploi :

- Alimenter la carte en appuyant sur le bouton poussoir rouge « ON ».
- La Led verte s'allume et s'éteint pour vérifier ce départ.
- (à faire la première fois) Réglages en liaison Bluetooth ou USB :
  - Choisir le calibre du courant (le même que celui de la pince).
  - Choisir la fréquence d'échantillonnage.
  - Choisir le numéro du fichier.
  - Éventuellement :
    - Étalonner l'intensité.
    - Étalonner la tension.
    - Régler l'horloge date et heure.
- Brancher la prise de tension sur l'appareil puis sur les batteries. La Led Rouge s'allume.
- Brancher la pince ampèremètrique sur l'appareil puis sur les batteries. La Led Jaune s'allume.
- Appuyer sur le bouton poussoir gris « START/STOP » pour l'enregistrement des courbes de décharge.

A partir de ce moment l'appareil mesure les tensions et intensités et les enregistre.
La Led verte clignote à a vitesse des échantillonnages.

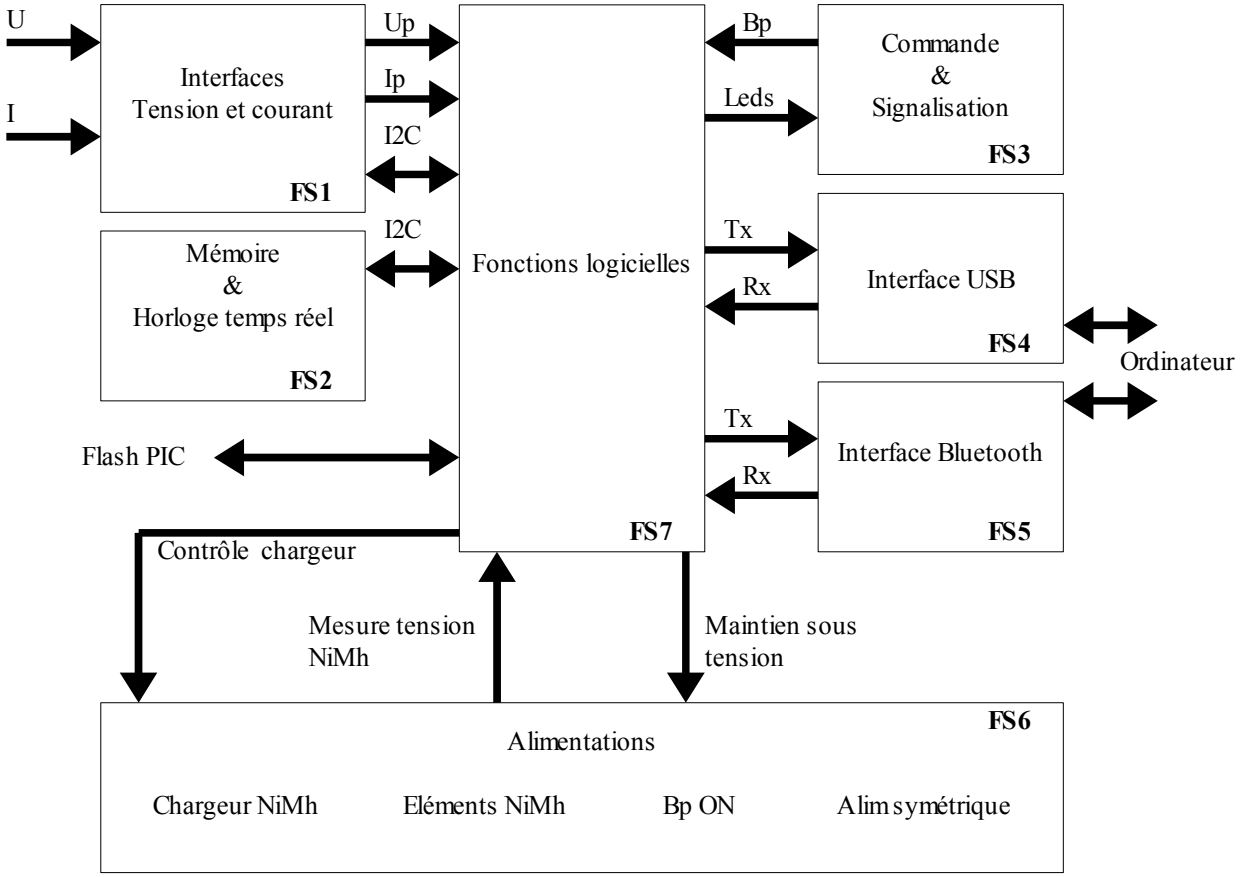
- A la fin des mesures, appuyer sur le bouton poussoir noir « START/STOP ». Si la mémoire est pleine l'appareil se comporte comme si on appuyait sur ce bouton pour arrêter les mesures.
- Déconnecter les prises de tension et de courant.
- En liaison Bluetooth ou USB :
  - Sur l'ordinateur lancer l'application « Serial Port Monitor » ou « Hyperterminal » et récupérer les données dans un fichier.
  - Lancer l'application « OpenOffice.org Calc » ou « Microsoft Excel », récupérer le fichier et construire le graphique de décharge.

On peut à tout moment éteindre l'appareil en maintenant le bouton poussoir gris appuyé pendant plus de 4 secondes. Une impulsion de la Led rouge marque l'arrêt.

Par liaison Bluetooth ou USB on peut observer l'appareil :
A la mise sous tension on a:
Date : 07/10/08 16:27:47
Echantillonnage: 1s
Calibre Pince: 100mV/A
Numero de fichier: 1
Pente_U: 1
Offset_U: 0
Pente_I: 1
Offset_I: 0
Quand on branche l'appareil on voit les valeurs de U et I :
U= 29 I= 99
Si on appuie sur ? on a l'aide des commandes :
M Envoie les mesures compatible Excel
I Etalonne l'intensité
U Etalonne la tension
R Remet toutes les mesures en mémoire à 0
C Calibre la pince ampèremétrique.
E Demande le temps d'échantillonnage
F Demande un numéro de fichier
H Mets l'horloge à jour
D Affiche l'horloge
B Tension NiMh
A => Arrêt appareil de mesure
Z Initialisation sans perte des données

On voit que :
Z redémarre un cycle.
A arrête l'alimentation.

Schéma fonctionnel :



FS1 : Interfaces Tension et Courant :
L'entrée tension va de -500 à +500V. Pour être sur d'avoir les ±400V.
L'entrée courant va de -150 à +150A. Pour être sur d'avoir les ±100A. Grâce à une pince ampèremètrique, c'est une tension pouvant aller de -15 à +15V qu'il faut prendre en considération.

FS2 : Mémoire et horloge temps réel :
La mémoire stocke les mesures.
L'horloge temps réel gère la date et l'heure.

FS3 : Commande et signalisation :
La commande permet de démarrer le cycle d'enregistrement.
La signalisation par 3 Leds permet de positionner l'appareil dans son cycle de travail.

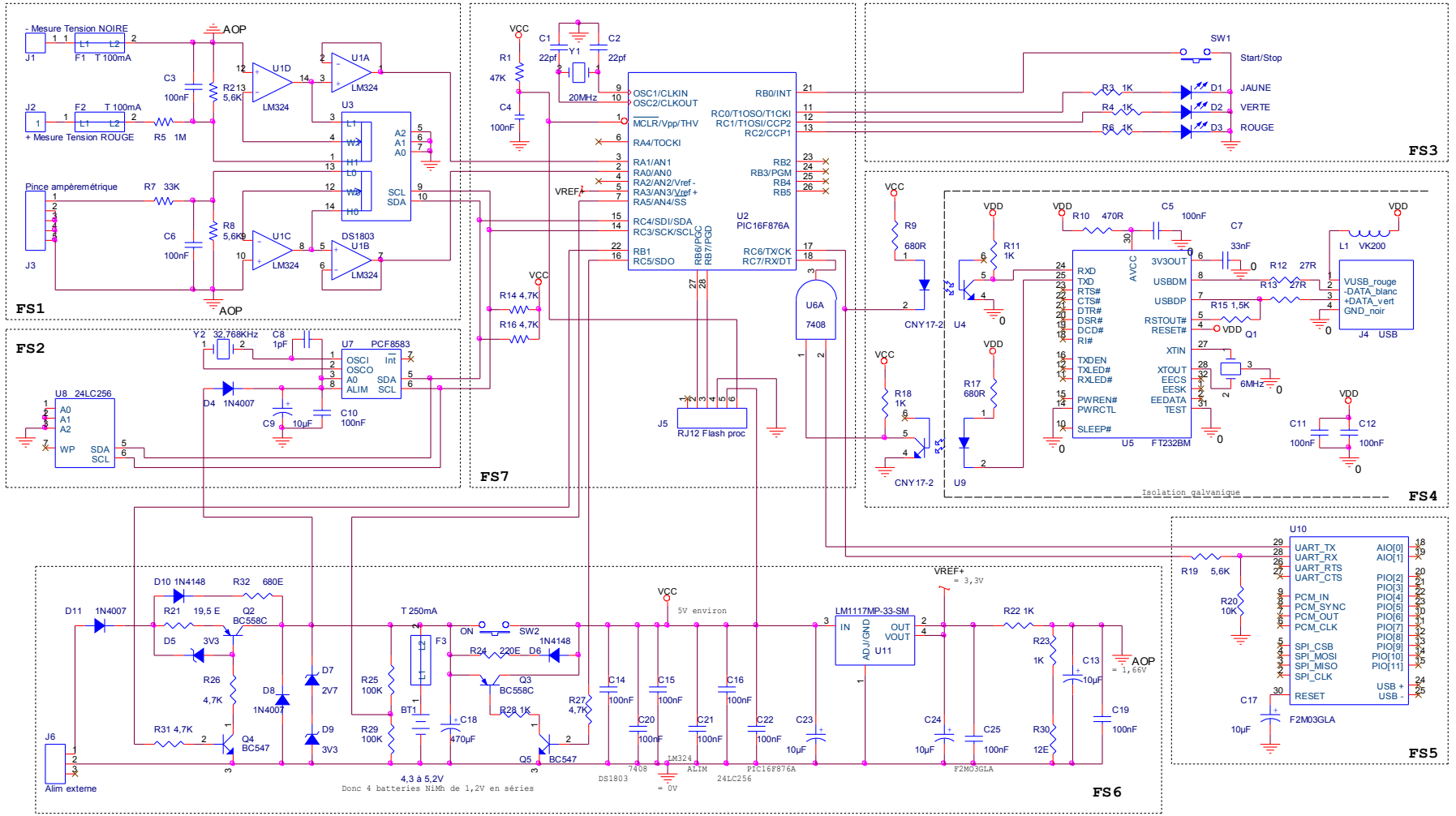
FS4 : Interface USB :
Par un cordon type A, un ordinateur peut être relié qui permettra de récupérer les mesures, d'étalonner l'appareil, de régler les paramètres : date, fréquence d'échantillonnage, ...

FS5 : Interface Bluetooth :
Un ordinateur équipé Bluetooth peut récupérer les mesures, étalonner l'appareil, régler les paramètres ...

FS6 : Alimentations :
Le chargeur gère la capacité en énergie des éléments NiMh.
Les éléments NiMh fournissent l'énergie électrique à l'appareil.
Le Bp « ON » met l'appareil sous tension.
L'alimentation symétrique alimente les amplificateurs opérationnels.

FS7 : Fonctions logicielles :
Elles seront expliquées par les algorithmes.

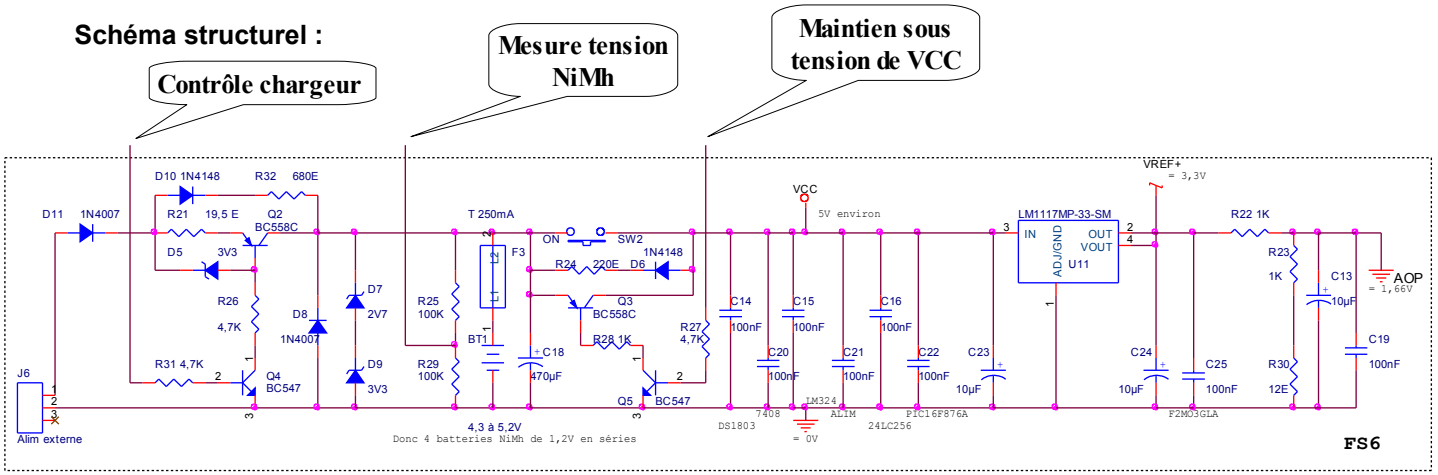
Schéma structurel :



Title	**Enregistreur autonome de décharge de batterie**	
Size B	Document Number **DAVID 2008**	Rev V5.5
Date: Monday, October 06, 2008	Sheet 1 of 1	

Alimentations :

Schéma structurel :



Bouton rouge « ON » :

Quand on appuie sur ce bouton on alimente toute la carte sauf l'horloge temps réel qui reste toujours sous tension. A ce moment là, le PIC16F876A sature le transistor Q5 qui sature à son tour le transistor Q3, ce qui court circuite le bouton poussoir rouge et maintient alors la mise sous tension des composants. Seul le PIC16F876A peut maintenant décider quand il faudra couper cette alimentation.

Tension d'alimentation :

La batterie NiMh « BT1 » connectée sur la carte, alimente le microcontrôleur, l'EEPROM externe, l'horloge temps réel, le potentiomètre numérique, le 74HC08, le module Bluetooth, et les amplificateurs opérationnels. Pour économiser l'énergie il est préférable de travailler directement avec la tension des batteries.
Le microcontrôleur ayant un oscillateur de fréquence 20MHz, il nécessite une tension comprise entre 4 et 5,5V.
La tension de référence haute Vref+ du convertisseur Analogique Numérique est limitée à VDD+0,3V (VCC+0,3V sur le schéma) et celle de référence basse Vref- à « Vref+ -2V ». L'EEPROM et l'horloge temps réel peuvent être alimentées de 2,5 à 5,5V. La tension de sortie de l'amplificateur opérationnel LM324 ne peut évoluer que de 0,02V à 3,3V pour une alimentation de 5V, mais qu'elle peut aller jusqu'à 32V. Le circuit 74HC08 fonctionne de 2 à 6V. Le module Bluetooth fonctionne sous 3,3V. En utilisant un régulateur de tension ayant une tension Vdrop maxi de 1,3V « entre entrée et sortie » il lui faut 3,3 + 1,3 = 4,6V mini pour fonctionner en toute sécurité. La tension max d'entrée du régulateur est de 20V.

Récapitulatif :

	Tension minimum en fonctionnement	Tension nominale maxi	Tension max à ne jamais dépasser
PIC16F876A	2V à 4MHz 3V à 10MHz 4,5V à 20MHz	5,5V	7,5V
24LC256I	1,8V	5,5V	7V
PCF8583	2,5V bus I2C actif 1V bus I2C inactif (veille)	6V	7V
DS1803	2,7V	5,5V	7V
LM324		5V	32V monotension -16 à +16V alim symétrique
74HC08	2V	6V	7V
Bluetooth	3V	3,3V	3,6Vmais 20V si régulateur
Conclusion. Limites de l'alimentation	**Au moins 4,5V**	**5,5V**	**7V maximum**

En conclusion il faut :

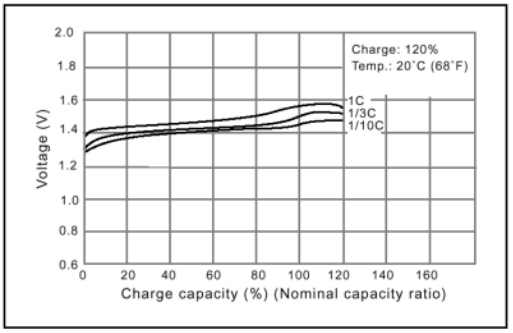
- Une alimentation VCC de 4,5 à 5,5V soit 4 éléments de 1,2V pour 4,8V.
- Une référence pour le CAN VREF+ = 3,3V, et on garde Vref- à 0V ce qui simplifie le schéma. (0,02V ≈ 0V).
- La masse virtuelle du LM324 : AOP=1,66V

Vérification des résistances du schéma :

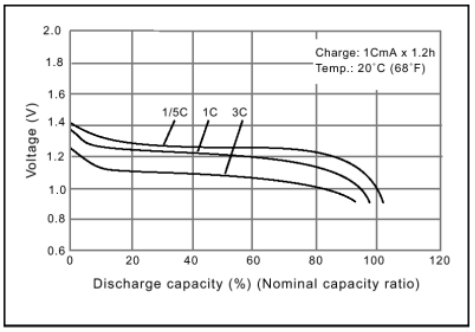
$$AOP = \frac{3,3 \times 1,012}{1+1,012} = 1,6598\,V$$ soit 1,66V

Surveillance de la charge et décharge des 4 éléments NiMh :

Sur les éléments NiMh, il est écrit :
Capacité 1900mAh.
Charge rapide 7 heures à 380mA.
Charge standard 16 heures à 190mA.



Les fabricants préconisent :
Le courant de charge rapide doit être compris entre 0,5 et 1 fois la valeur du 1/10ème de la capacité en mA.h, soit de 95 à 190mA.
La détection de la tension maximale est à environ 1,8V/élément, soit 7,2V pour 4 éléments.
Courant de charge permanente 1/30ème à 1/20ème du 1/10èmè de la capacité en mA.h, soit de 6,5 à 9,5 mA.
Limiteur de temps de la charge rapide : 90 min
Limiteur de temps total de charge : 10 à 20 heures.



Le choix :
Il faut que le chargeur soit simple aussi le principe utilisé sera de charger à un courant constant de 140mA et de ne pas dépasser les limites de tension 4,6 à 5,5V.
Vérification :
La tension au moment de la charge est supérieure à celle de la décharge pour la même capacité.
Sur la courbe de charge, pour une capacité de 50%, la tension vaut de 1,4 à 1,5V tandis que pour la décharge elle vaut de 1,1 à 1,3V. Ceci fait que la tension peut varier de 4,4 à 6V.
Pour une capacité de 100% la tension varie de 1V (il ne faut pas descendre en dessous) à 1,55V soit de 4 à 6,2V
On est dans les limites absolues, il faut bien surveiller le chargeur. **On autorisera de 4,5 à 6V.**

Mesure de la tension des 4 éléments NiMh :
Comme les résistances R25 et R29 sont identiques et égales à 100K, elles permettent de prélever pour l'entrée du PIC16F876A la moitié de la tension des 4 éléments NiMh. (6V maxi).

La charge se fait grâce au connecteur d'entrée et au transistor Q2 (piloté par Q4) monté en régulateur de courant.
I_{Q2} est réglé par la résistance R21 et la zener D5.

$$I = \frac{Vz - 0,6}{R21} = \frac{3,3-0,6}{19,5} = 139\,mA$$ Pour une capacité de 2700mAh (le max actuellement pour des LR6) c'est un peu moins que le dixième de la capacité pour le courant de charge, et pour des 1900mAh (bas de gamme) c'est presque le dixième de la capacité. Le chargeur ne fonctionne que si le microcontrôleur est alimenté.

Limitation de tension :

Si on déconnecte les éléments NiMh, grâce au transistor Q2, le chargeur peut faire monter la tension d'alimentation. La limitation est faite par 2 zeners de 2,7+3,3=6V. Le courant étant de 139mA, la puissance à dissiper est donc de 139*6=834mW. Les zeners doivent pouvoir dissiper 1/2W.

Charge d'entretien :

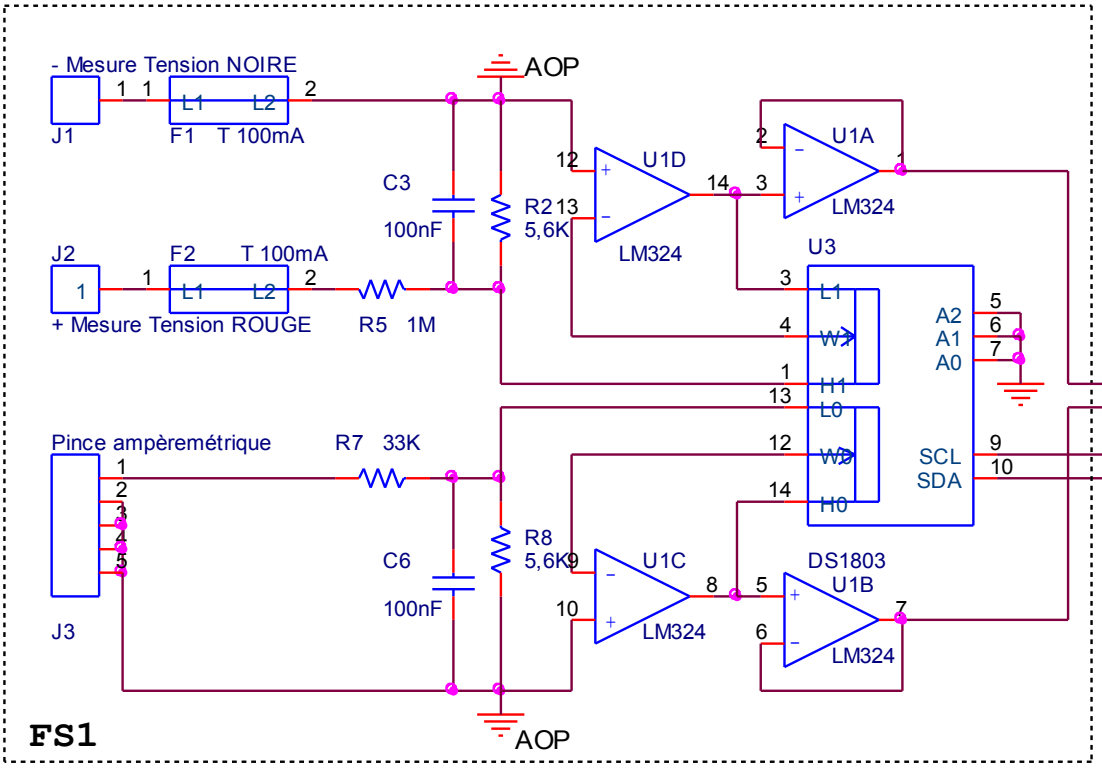
D10 et R32 permettent de fournir un courant de l'ordre de 10mA aux batteries quand l'appareil est relié au secteur pour les maintenir chargées.

Anti retour :

La diode D11 a été ajoutée parce qu'au moment de la connexion avec un bloc secteur, un courant inverse circulait depuis la batterie vers la charge du condensateur de ce bloc secteur. Le fusible n'appréciait pas.

Interfaces tension et courant :

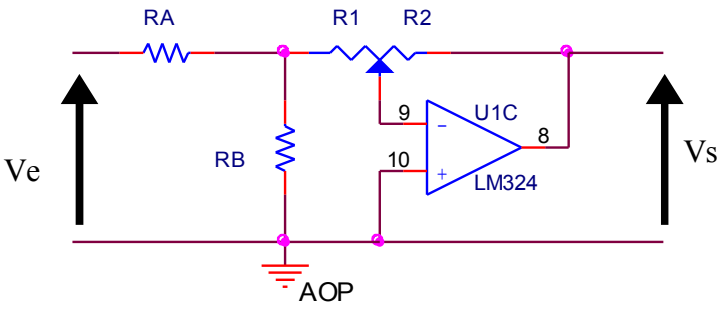
Schéma structurel :



Étude commune aux 2circuits :

Le même schéma est utilisé pour les mesures de tension et de courant. Les potentiomètres numériques sont réglés par une information provenant du bus I2C. Le schéma structurel peut se résumé à l'extrait ci-contre.

Pour calculer le Gain en tension Vs/Ve de ce montage, une solution consiste à remplacer RA et RB par un générateur de Thévenin. C'est le schéma d'après.



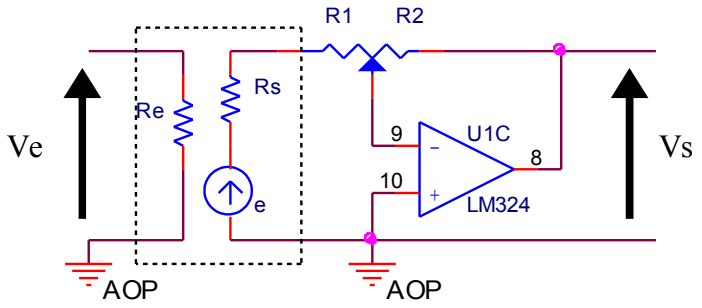
$$Rs = RA \,//\, RB = \frac{RA \times RB}{RA + RB}$$

$$e = Ve \times \frac{RB}{RA + RB}$$

$$\frac{e}{Ve} = \frac{RB}{RA + RB}$$

$$G = \frac{Vs}{Ve} = \frac{Vs}{e} \times \frac{e}{Ve}$$

$$G = - \frac{R2}{R1 + Rs} \times \frac{RB}{RA + RB}$$



$$G = - \frac{R2}{R1 \times (1 + \frac{RA}{RB}) + RA}$$

Si N est la valeur numérique qui fixe la position du curseur du potentiomètre ($0 \le N \le 255$) et P = 100KΩ sa résistance ohmique, on a $R1 = \frac{P \times N}{256}$ et $R2 = P - \frac{P \times N}{256}$

Donc suivant les besoins on utilisera :

$$\frac{Vs}{Ve} = -\frac{256 - N}{N \times (1 + \frac{RA}{RB}) + 256 \times \frac{RA}{P}} \quad \text{ou} \quad N = 256 \times \frac{1 + \frac{Vs}{Ve} \times \frac{RA}{P}}{1 - \frac{Vs}{Ve} \times (1 + \frac{RA}{RB})}$$

Mesure du courant :

Pour mesurer les courants positifs ou négatifs « I » de 150A on utilise une pince ampèremètrique à effet hall dont le calibre en mV/A s'appellera « C ». Donc $Ve = I \times C$

La tension de 1,66V se trouve à mi-distance entre 0,02 et 3,3V. Cette tension s'appelle masse AOP dans le schéma structurel. Pour simplifier les calculs on dira que c'est une masse, que la tension d'alimentation est symétrique et de valeur ±1,64V. La sortie de l'AOP varie donc de -1,64 à +1,64V pour +150A à -150A (ampli inverseur) quel que soit le calibre de la pince.

Par contre la tension à l'entrée du Convertisseur Analogique Numérique (CAN) du microcontrôleur « Vp » est référencée par rapport à la masse qui est décalée de 1,66V de la masse AOP.

Le nombre « Nb » recueilli à la sortie du CAN est codé sur 10 bits. Nb=0 pour Vp=0V et Nb=1023 pour Vp=3,3V.

Si on part de $N = 256 \times \frac{1 + \frac{Vs}{Ve} \times \frac{RA}{P}}{1 - \frac{Vs}{Ve} \times (1 + \frac{RA}{RB})}$ avec $Ve = I \times C$ et en remplaçant les résistances par leur valeur, RA=R7=33KΩ, RB=R8=5,6KΩ et P=100KΩ, on peut régler le curseur du potentiomètre N en fonction du calibre C : $N = 256 \times \frac{C - 3{,}63}{C + 75{,}8}$

Le tableau donne des exemples :

Courant dans la pince ampèremètrique	Calibre de la pince	Tension de la pince	Gain de l'AOP	Tension en sortie de l'AOP par rapport à la masse AOP.	Tension entrée PIC par rapport à la masse GND	Nombre à la sortie du CAN	Variable dans la fonction de mesure du courant
I	C	Ve	G	Vs	Vp	Nb	I
-150A	10mV/A	-1,5V	1,093	1,64V	3,3V	1023	-150
-150A	100mV/A	-15V	0,1093	1,64V	3,3V	1023	-150
0A	10mV/A	0V	1,093	0V	1,66V	512	0
150A	100mV/A	15V	1,093	-1,64V	0,02V	6 (presque 0)	150

La valeur calculée par la fonction en langage C de la mesure du courant doit redonner l'intensité mesurée par la pince.

- Pour I=150A on a Nb=0
- Pour I=0A on a Nb=512
- Pour I=-150A on a Nb=1023

Donc $I = 150 - \frac{300}{1024} \times Nb$ soit $I = 150 \times (1 - \frac{Nb}{512})$

Pour faire un réglage de l'amplification on peut utiliser une variable Pente I. C'est **l'étalonnage de la pente**.

Pour régler le 0 il faut que NB/512=1. Si NB n'est alors pas à 512 il faut le modifier, ajouter ou retrancher une valeur. C'est **l'étalonnage du décalage**.

L'équation entrée dans le programme sera donc : $I = Pente\ I \times 150 \times (1 - \frac{Nb + Offset\ I}{512})$

A l'initialisation, Pente_I vaut 1 et offset_I vaut 0 (nombre signé).

Nombre de bits disponibles sur le CAN pour la mesure de précision :
Pour 3,3V en mode 10 bits on a 1024 pas;
Pour 300A de variation on a 3,3 – 0,02 = 3,28V utilisables soit (1024*3,28)/3,3 = 1018 bits utilisables.
Le quantum est donc de 300/1018 ≈ 0,295A par incrément du CAN.
Cette précision de mesure devrait être suffisante.
L'organisation du microcontrôleur oblige à utiliser A0, A1, A2 et A5 comme entrée analogique avec A3 comme borne de Vref+.

Mesure de la tension:

En partant du gain :

$$\frac{Vs}{Ve} = -\frac{256-N}{N\times(1+\frac{RA}{RB})+256\times\frac{RA}{P}}$$ et pour Ve = la tension à mesurer qui peut aller de -500 à 500V, on peut trouver Ve = f(N,Nb).

$$Ve = -\frac{(179{,}57\times N + 2560)\times(\frac{3{,}3\times Nb}{1024}-1{,}65)}{256-N}$$

Quand on lit Nb sur le convertisseur, un algorithme ajuste N ce qui permet de n'être ni en saturation ni en trop faible valeur.
Pour l'étalonnage du 0, comme pour la mesure de l'intensité il faut modifier NB pour que le numérateur de l'équation fasse 0.
Pour l'étalonnage de la tension maximum, c'est la pente de l'équation qu'il faut ajuster. Il faut donc multiplier l'équation par un nombre en pourcent pour faire l'ajustement.

L'équation devient donc : $$Ve = -Pente\,U\times\frac{(179{,}57\times N + 2560)\times(\frac{3{,}3\times(Nb+Offset\,U)}{1024}-1{,}65)}{256-N}$$

Pente_U tourne autour de 1. Par exemple il peut varier et valoir 0,999 ou 1,001 pour diminuer ou augmenter la tension de 1‰. C'est **l'étalonnage de la pente**.
Offset_U vaut autour de 0. Par exemple il peut prendre les valeurs +1 ou -1. C'est **l'étalonnage du décalage**.

Potentiomètre numérique DS1803 :

L'octet de contrôle est : 0 1 0 1 A2 A1 A0 R/W.
Dans le programme ce sera : A0=A1=A2=0V => 0x50 écriture => 0X51 lecture
Les octets de commande sont :

COMMAND	COMMAND VALUE
Write Potentiometer-0	101010 01
Write Potentiometer-1	101010 10
Write Both Potentiometers	101011 11

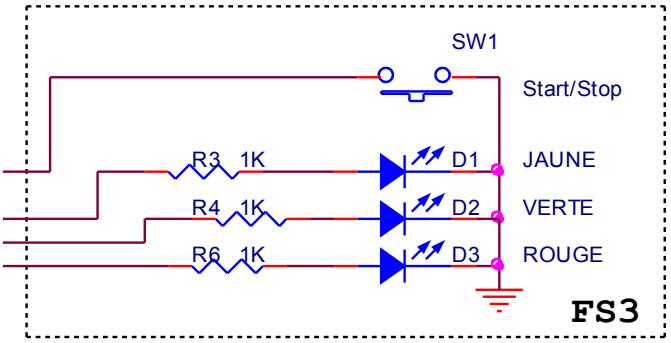
La trame à envoyer est : octet de contrôle, octet de commande, donnée.

Remarque :
Dans le schéma structurel les bornes L et H ne sont pas raccordées de la même façon pour la tension ou l'intensité. Il y aura donc une différence dans la programmation.

Commande et signalisation :

Schéma structurel :

Le portB du microcontrôleur dispose de résistances internes de tirage vers le haut.
Donc quand la broche n'est pas raccordée à un potentiel, elle vaut 1 logique. Quand on actionne le bouton poussoir on met RB0 du microcontrôleur à la masse ce qui provoque un état logique 0.

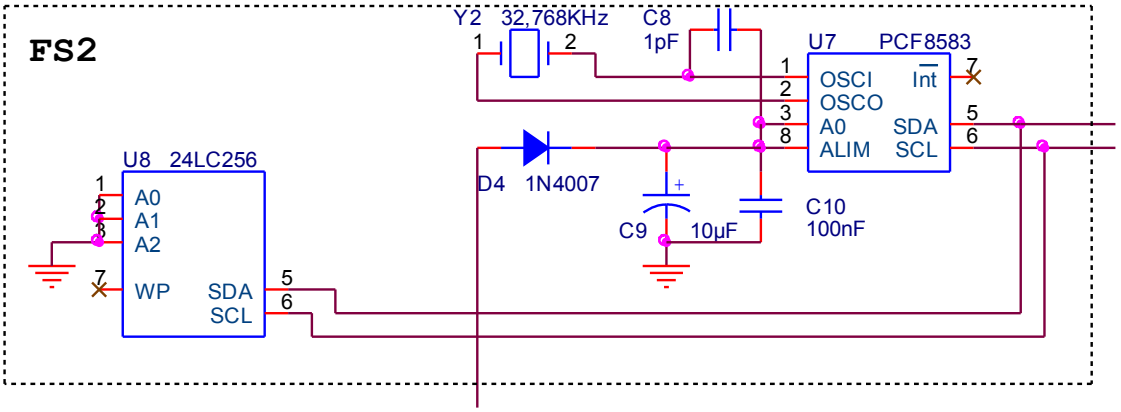


Les Leds sont à haute luminosité et à faible consommation. Le courant vaut

$$I = \frac{(4{,}5-2)}{1\mathrm{K}} = 2{,}5\,mA$$

Mémoire et horloge temps réel :

Schéma structurel :



EEPROM externe 24LC256 :

L'octet de contrôle est : 1 0 1 0 A2 A1 A0 R/W.
Dans le programme ce sera : A0=A1=A2=0V => 0xA0 écriture => 0XA1 lecture.
La trame à envoyer ou recevoir est :
octet de contrôle, adresse EEPROM poids fort, adresse EEPROM poids faible, donnée.

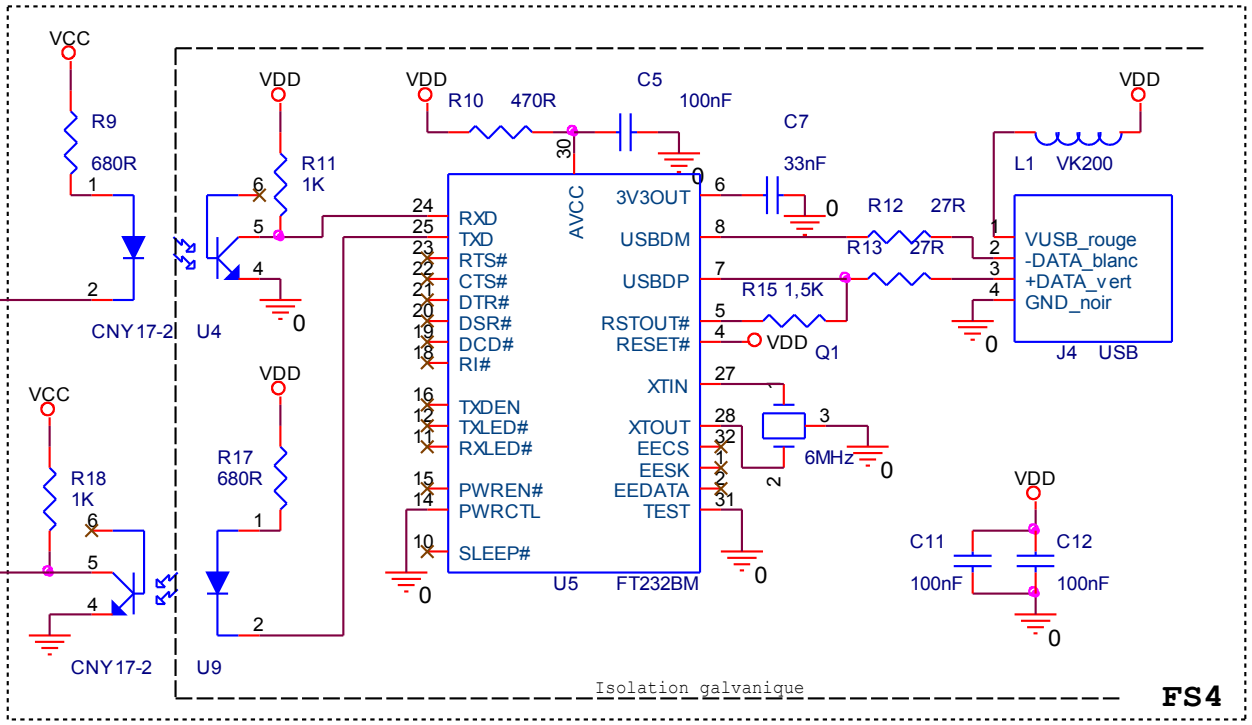
Horloge temps réel PCF8583 :

L'octet de contrôle est : 1 0 1 0 0 0 A0 R/W.
Dans le programme ce sera : A0=+5V => 0xA2 écriture => 0XA3 lecture.
La trame à envoyer ou recevoir est : octet de contrôle, adresse , donnée.

L'alimentation du circuit est prise directement sur la batterie pour ne pas être interrompue à la mise hors tension. La diode D4 et le condensateur C9 garantissent une réserve d'énergie sur un petit laps de temps.

Interface USB :

Schéma structurel :



Le schéma est tiré d'une application du constructeur qui a été simplifiée.
Comme des tensions de 500V peuvent se présenter sur l'entrée de l'enregistreur, il faut une isolation galvanique avec l'ordinateur qui sera raccordé. Les photocoupleurs CNY17-2 permettent cette isolation et l'ordinateur alimentera le circuit FT232BM par du 5V. Pour les 2 sens de circulation du signal :
Quand :

- Le Tx =0 logique, il vaut 0V et la Led du photocoupleur est allumée.
- Le Tx =1 logique, il vaut 5V et la Led du photocoupleur est éteinte.

Quand :

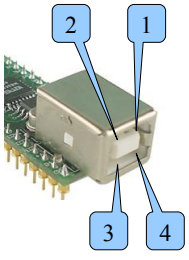
- Le transistor est éclairé, il est passant; le collecteur vaut 0V donc Rx = 0 logique.
- Le transistor n'est pas éclairé, il est donc bloqué; le collecteur vaut 5V donc Rx = 1 logique.

En conclusion quand :

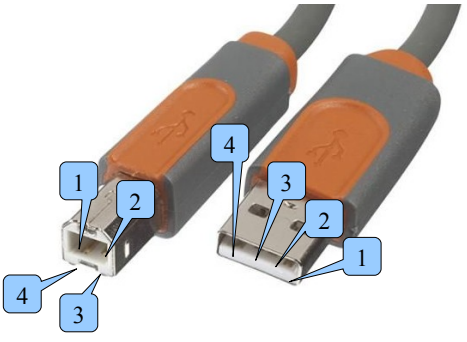
- Tx=0 logique => Rx = 0logique.
- Tx=1 logique => Rx = 1logique.

Sur l'ordinateur il faut un "driver" qui va ouvrir un port série virtuel COM compatible RS232. On peut trouver ce "driver" à l'adresse du constructeur FTDI : http://www.ftdichip.com/FTDrivers.htm

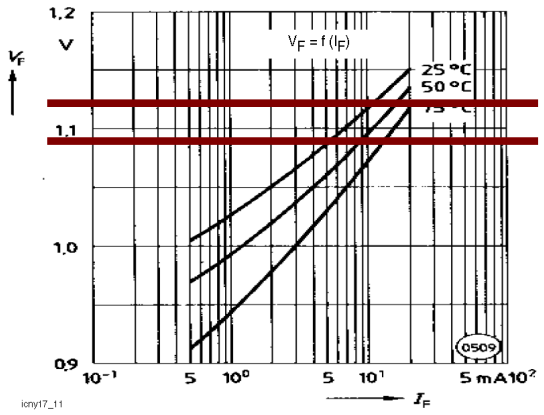
Brochages des connecteurs :

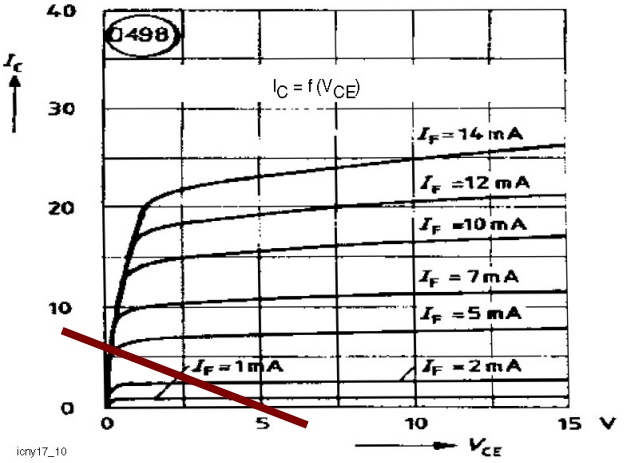


+5V	Data-	Data+	Masse
1	2	3	4
Rouge	Blanc	Vert	Noir



Calculs autour du photocoupleur :





Le graphique de droite montre que les courants de I_F vont raisonnablement de 2 à 14 mA. 5mA semble un choix correct si on veut économiser l'énergie.
Le graphique de gauche dit alors que Vf vaut 1,09V environ pour 5mA à 25°C. Donc la résistance qui limite le courant dans la Led vaut :

$$R = \frac{U}{I} = \frac{5-1,09}{5\times10^{-3}} = 782$$ On prendra **680Ω.**

La droite de charge du transistor doit donner 0V pour I_F = 5mA et +5V pour I_F =0mA. Donc la résistance vaut :

$$R = \frac{5}{5\times10^{-3}} = 1\text{K}$$

Interface Bluetooth :

Schéma structurel :

L'interface est gérée par le module F2MO3GLA de Free2move..
L'alimentation du module est de 3,3V.

Sur l'entrée UART_RX la tension correspondant au 1 logique est de 3,3V tandis que pour le 0 logique elle est de 0V. C'est le rôle du pont diviseur de tension R19 et R20.
Pour la sortie UART_TX le 0 logique vaut 0V, mais le 1 logique vaut 3,3V. Ceci reste compatible avec l'entrée du microcontrôleur qui accepte cette tension comme 1 logique.
Le module de base est un circuit imprimé :
Son soudage sur la carte imprimée est délicate, c'est vraiment sa seule difficulté, car la mise en oeuvre est très simple. Pour le programmer il faut le connecter sur l'interface série d'un ordinateur.

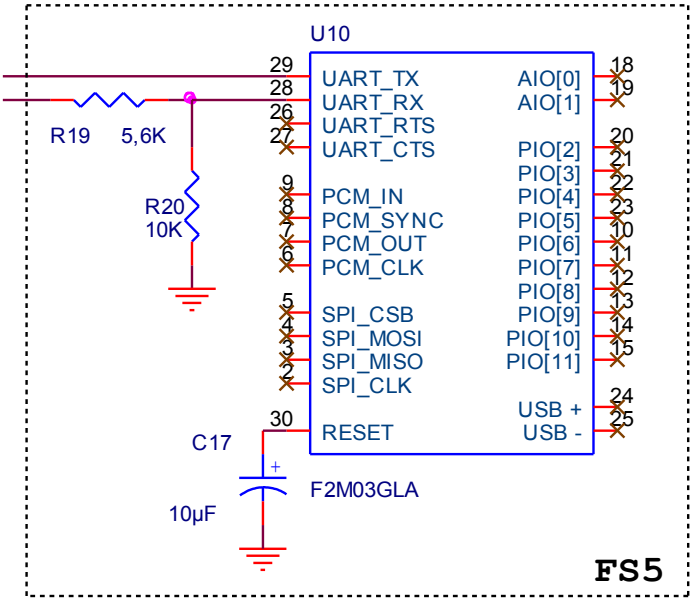
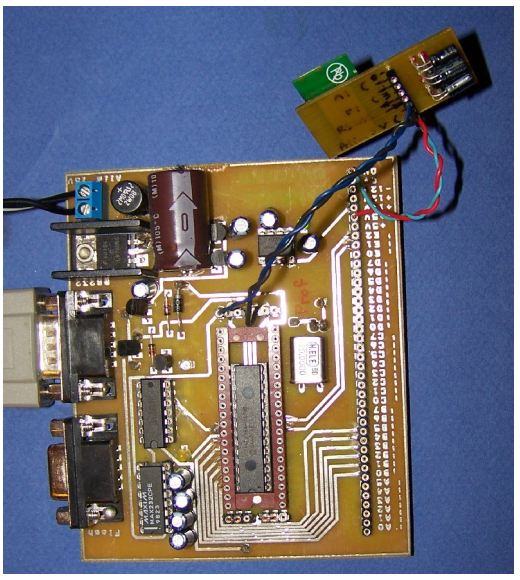


Schéma de raccordement de la carte sur un ordinateur :

Le module soudé sur la carte permet de faire travailler la liaison série sous 5V. Pour le relier sur le port série d'un ordinateur, il faut ajouter une interface RS232 qui converti les tensions en +12V et -12V. Ce peut être fait en fil volant sur une platine d'essais ou en détournant une application comme le montre cette photo. C'est le schéma classique qui utilise un MAX232 ou un MAX233.
Ce raccordement permettra de configurer le module ou de le tester en autonome sur un ordinateur.



Procédure de raccordement de la carte sur l'interface ordinateur :

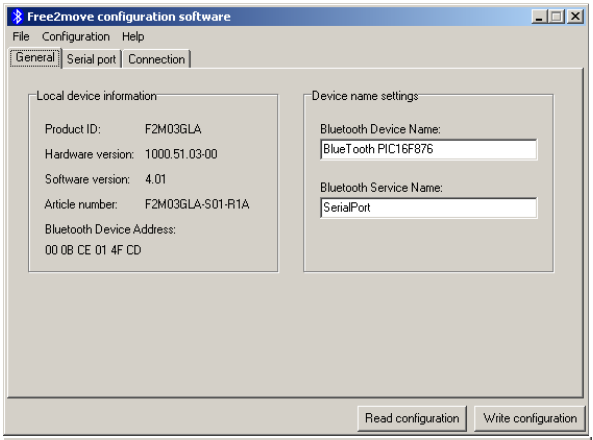
- Enlever le circuit U4 (CNY17-2).
- Enlever le circuit U6 (74LS08).
- Enlever le circuit U2 (PIC16F876A).
- Raccorder la broche 17 de U2 (Rx du F2M03GLA) à la broche Tx du COM:x de l'ordinateur après interface par MAX232.
- Raccorder la broche 2 de U6 (Tx du F2M03GLA) à la broche Rx du COM:x de l'ordinateur après interface par MAX232.

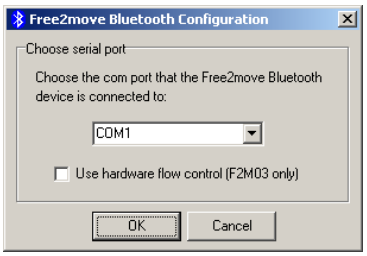
On peut utiliser la carte MiniPicRs232.

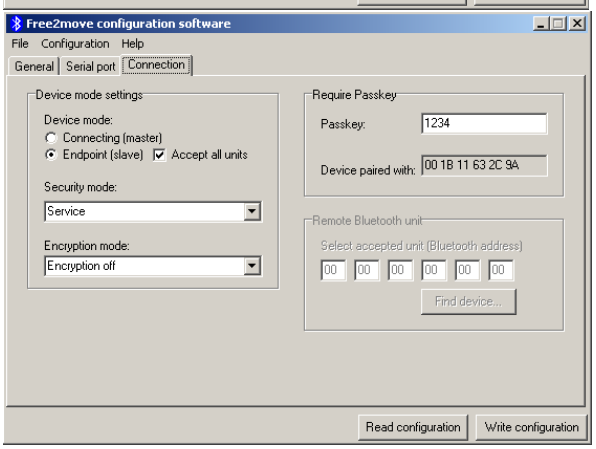
Changer la photographie

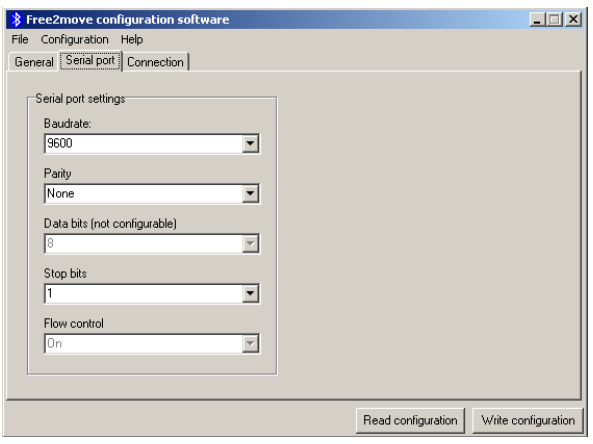
Free2move, configuration du module :

C'est le logiciel qui permet de configurer le module F2M03GLA. Il faut utiliser le fichier 33_1198234843_F2M_BlueCFG_4.exe qui exécute la version 4.06.00.
Les images suivantes montrent les critères de configuration qu'il faut utiliser. On valide sur COM1 si le port série de l'ordinateur utilisé est le COM1 bien évidement.



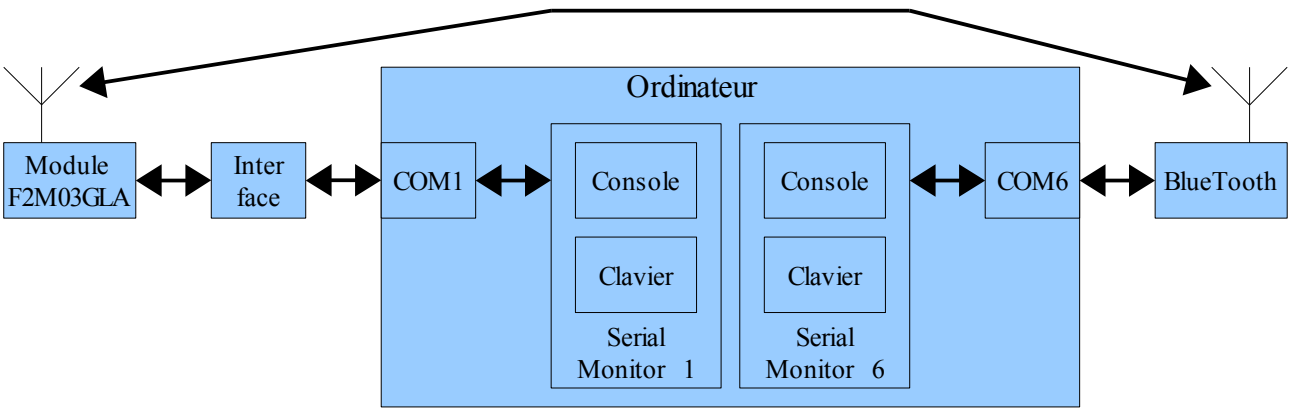






Test de la liaison :

Pour tester le montage il faut disposer d'un ordinateur ayant une liaison Bluetooth ou sur lequel on a ajouté une clé du style USB-Bluetooth. Il faut aussi avoir un logiciel de dialogue avec la liaison série du style hyperterminal fourni par Windows. Personnellement je préfère l'outil du compilateur PICC (PCW compiler) « Serial Port Monitor » qui est directement utilisable sans aucune espèce de configuration abracadabrante.



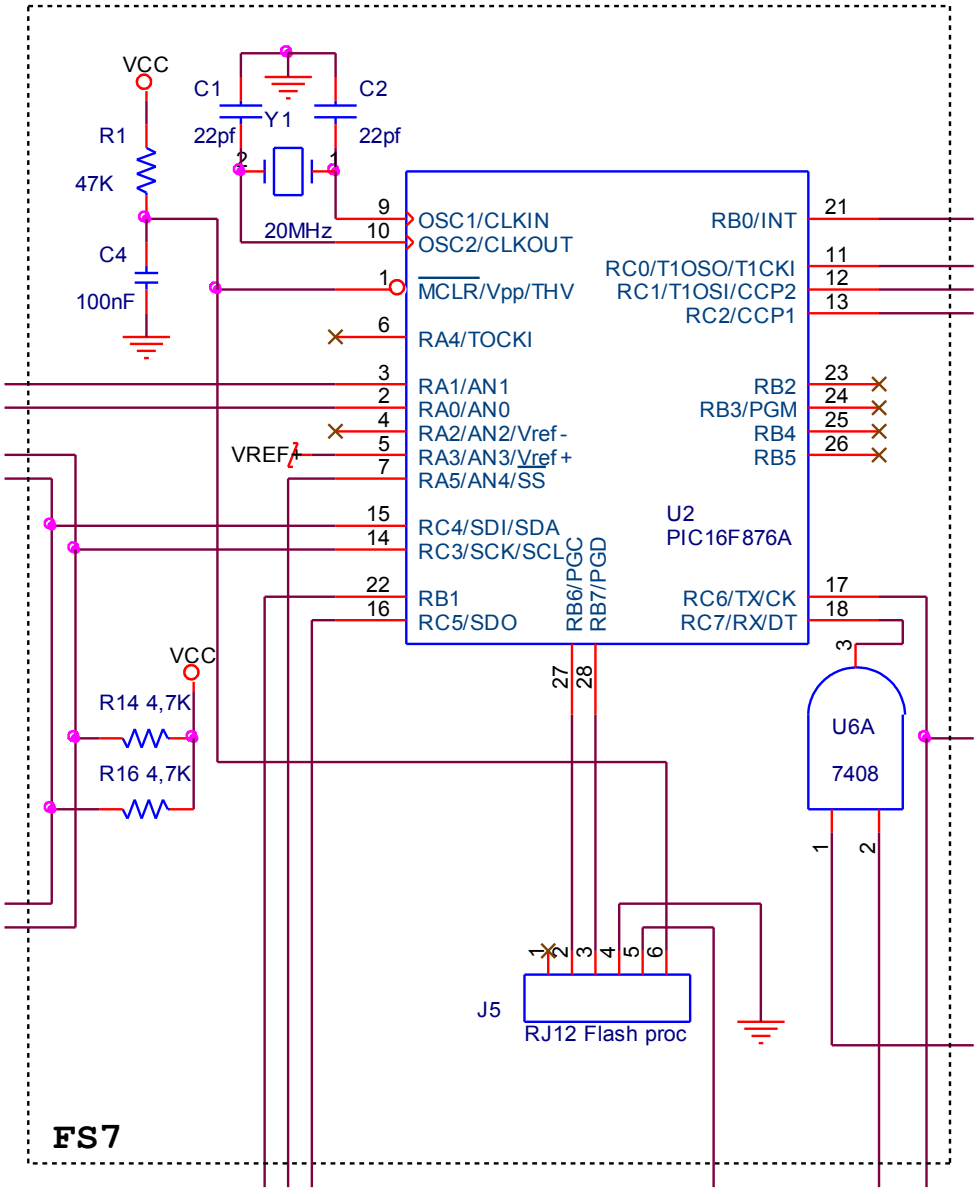
Comme le schéma bloc le montre il faut :

- Relier électriquement le module F2M03GLA, broches Rx Tx sur une interface du type MAS232 et entrer sur le port COM1 de l'ordinateur et mettre le module sous tension.
- Démarrer la liaison Bluetooth de l'ordinateur.
- Démarrer 2 fois le logiciel Serial Monitor.
- Configurer le port COM1 sur le 1er Serial Monitor
- Configurer le port COM xx dont le numéro dépend de votre ordinateur et qui est provoqué par la réception du F2M03GLA par le Bluetooth de l'ordinateur. Pour l'exemple on a dit que c'était le port COM6.

Le principe de cette liaison complexe consiste à écrire sur le clavier de Serial Monitor 1 et de recevoir le texte sur Serial Monitor 6.
On peut aussi écrire sur le clavier de Serial Monitor 6 et recevoir sur la console 1. Le trajet du texte se fait par liaison Bluetooth, par les ondes radios.

Fonctions logicielles :

Schéma structurel :



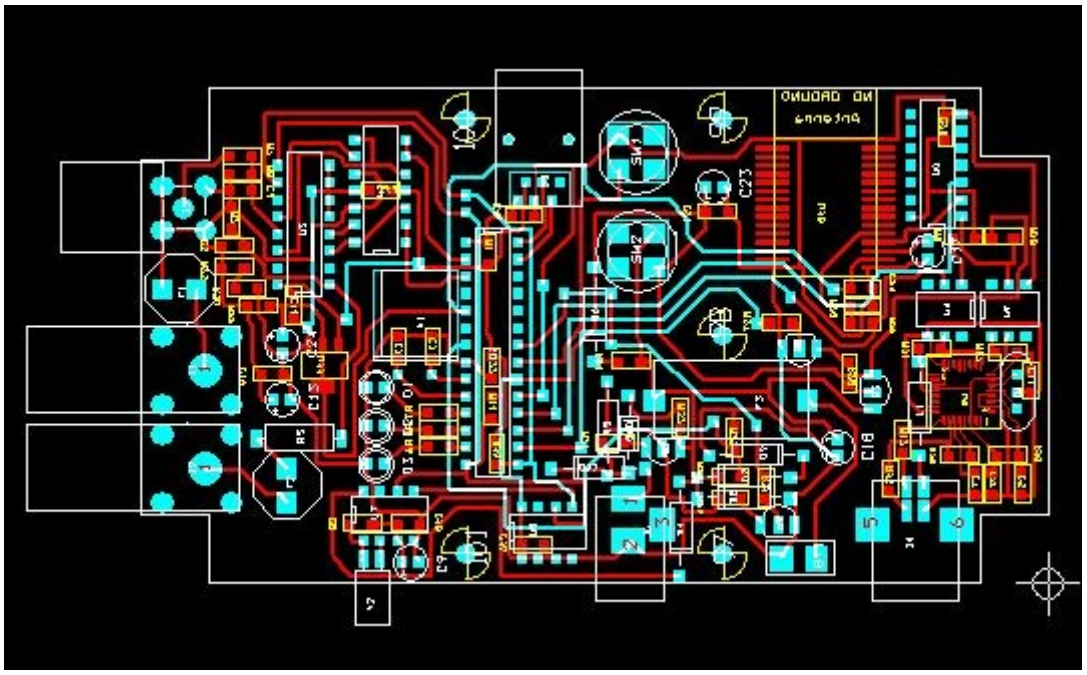
Pour utiliser 3 entrées analogiques avec une référence à +3,3V il faut se connecter sur les broches AN0, AN1, AN2 et AN4 pour les entrées analogiques et sur AN3 pour la référence de tension de 3,3V. Dans le montage l'entrée AN2 n'est pas utilisée.

- AN0 mesure la tension.
- AN1 mesure l'intensité.
- AN4 mesure la tension des batteries de l'alimentation de la carte.
- Vref+ AN3 est la référence tension à +3,3V.
- Vref- n'est pas utilisée. Le PIC considère que Vref- est à la masse.

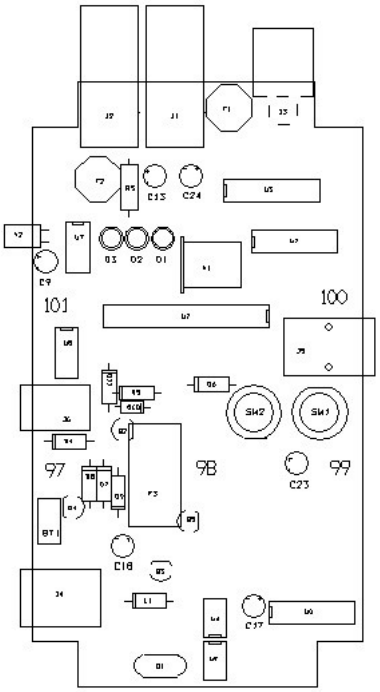
Toutes les autres entrées ou sorties sont logiques.
L'utilisation de la liaison RS232 impose un quartz sur l'oscillateur du microcontrôleur afin d'avoir suffisamment de précision dans les échanges.
Le **7408 est un 74HC08**, il est l'aiguillage du Rx depuis le Bluetooth ou depuis l'USB.

Reprogrammation du PIC16F876A :
Le connecteur J5 permet de se connecter sur une carte MiniPicRs232 pour charger le programme dans la mémoire flash du PIC16F876A. Il permet aussi une mise sous tension ce qui impose l'envoie d'énergie dans les éléments NiMh et l'horloge temps réel, même si le poussoir rouge n'est pas appuyé. Ceci est fait grâce à D6 et R24. Tous les composants seront donc alimentés.

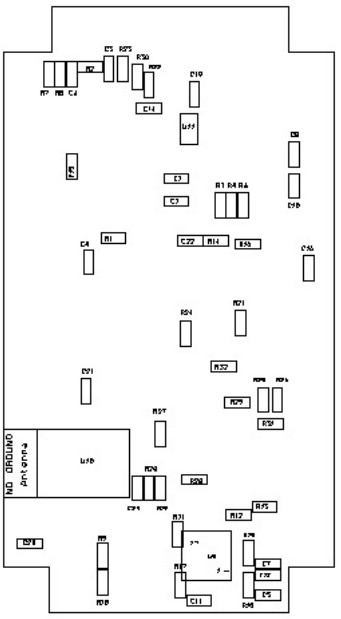
Circuit imprimé :



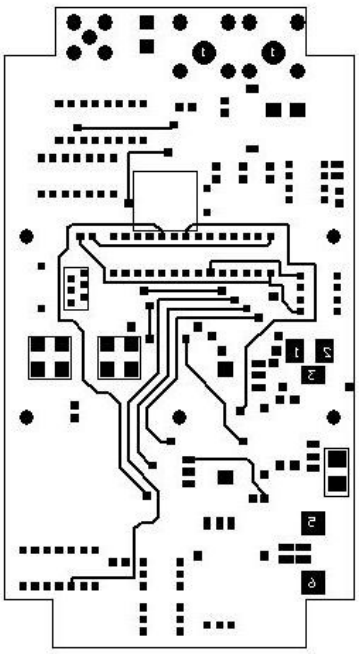
Composants coté composants



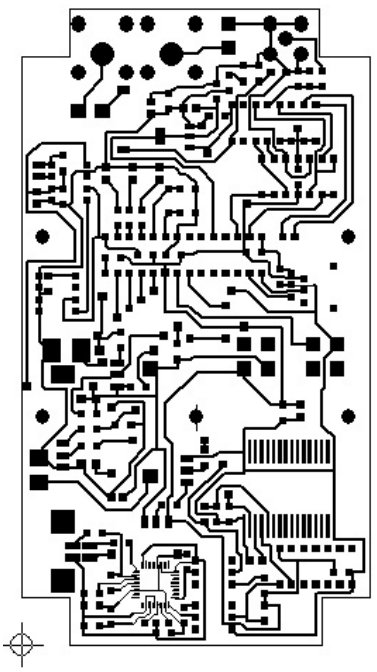
Composants coté cuivre



Circuit imprimé coté composants



Circuit imprimé coté cuivre



Bon de commande :

Enregistreur autonome de décharge de batterie Revised: Monday, June 16, 2008 F=Farnell R=Radiospare
DAVID 2008 Revision: V5.5 S=Selectronic
Bill Of Materials June 17,2008 22:12:39 Page1 E=Electronique diffusion

Item	Quantité	Référence	Part	Nom et code fournisseur	Prix Unitaire HT	Montant Total HT
1	1	BT1 4,3 à 5,2V	Coupleur piles 4 LR6			0,00 €
2	2	C1,C2	22pf cms 1206			0,00 €
3	15	C3,C4,C5,C6,C10,C11,C12,C14,C15 ,C16,C19,C20,C21,C22,C25	100nF cms 1206			0,00 €
4	1	C7	33nF cms 1206			0,00 €
5	1	C8	1pF cms 1206			0,00 €
6	5	C9,C13,C17,C23,C24	10µF 25V chimique			0,00 €
7	1	C18	470µF 25V chimique			0,00 €
8	1	D1	Led Jaune			0,00 €
9	1	D2	Led Verte			0,00 €
10	1	D3	Led Rouge			0,00 €
11	3	D4,D8,D11	1N4007			0,00 €
12	2	D5,D9	Zener 3V3			0,00 €
13	2	D6,D10	1N4148			0,00 €
14	1	D7	Zener 2V7			0,00 €
15	2	F1,F2	Fusible T 100mA			0,00 €
16	1	F3	Fusible T 250mA			0,00 €
17	1	J1	Borne 4mm NOIRE			0,00 €
18	1	J2	Borne 4mm ROUGE			0,00 €
19	1	J3	BNC			0,00 €
20	1	J4	Connecteur USB A sur CI			0,00 €
21	1	J5	Connecteur flash PIC			0,00 €
22	1	J6	Bornier Jack 2mm			0,00 €
23	1	L1	VK200			0,00 €
24	1	Q1	Résonnateur 6MHz			0,00 €
25	2	Q2,Q3	BC558C			0,00 €
26	2	Q4,Q5	BC547			0,00 €
27	1	R1	47K cms 1206			0,00 €
28	8	R3,R4,R6,R11,R18,R22,R23,R28	1K cms 1206			0,00 €
29	3	R2,R8,R19	5,6K cms 1206			0,00 €
30	1	R5	1M traversant			0,00 €
31	1	R7	33K cms 1206			0,00 €
32	2	R17,R9	680R cms 1206			0,00 €
33	1	R10	470R cms 1206			0,00 €
34	2	R12,R13	27R cms 1206			0,00 €
35	5	R14,R16,R26,R27,R31	4,7K cms 1206			0,00 €
36	1	R15	1,5K cms 1206			0,00 €
37	1	R21	19,5 E cms 1206			0,00 €
38	2	R25,R29	100K cms 1206			0,00 €
39	1	R20	10K cms 1206			0,00 €
40	1	R24	220 E cms 1206			0,00 €
41	1	R30	12 E cms 1206			0,00 €
42	1	R32	680E cms 1206			
43	1	SW1	Start/Stop			0,00 €
44	1	SW2	ON			0,00 €
45	1	U1	LM324			0,00 €
46	1	U2	PIC16F876A			0,00 €
47	1	U3	DS1803			0,00 €
48	2	U4,U9	CNY17-2			0,00 €
49	1	U5	FT232BM			0,00 €
50	1	U6	74HC08			0,00 €
51	1	U7	PCF8583			0,00 €
52	1	U8	24LC256			0,00 €
53	1	U10	F2M03GLA			0,00 €
54	1	U11	LM1117MP-33-SM			
55	1	Y1	20MHz			0,00 €
56	1	Y2	32,768KHz			0,00 €
57	0,71 dm²	Circuit imprimé double face	8,2 × 8,6 cm	Imprelec		0,00 €
58	1		Support 28 broches			0,00 €
59	1		Support 16 broches			0,00 €
60	2		Support 14 broches			0,00 €
61	2		Support 8 broches			0,00 €
62	2		Support 6 broches			0,00 €
63	1		Cordon USB A-B			0,00 €
64			Petit matériel			0,00 €
65			Main d'oeuvre			0,00 €
66			Boîtier			0,00 €

Planning d'étude et de réalisation :

Dates limites de la semaine					**1ère revue de projet**					**2ème revue de projet**					**3ème revue de projet**
Rang semaine	**1**	**2**	**3**	**4**		**5**	**6**	**7**	**8**		**9**	**10**	**11**	**12**	
Réception et étude du cahier des charges Étude fonctionnelle Étude du mode d'emploi															
Étude du schéma structurel sur OrCad Calcul du dimensionnement des composants															
Simulation du schéma sur logiciel Test des fonctions électroniques sur platine d'essais Recherche des revendeurs, prix et références des composants Édition du bon de commande															
Création d'une bibliothèque personnelle sur OrCad Layout. Recherche ou création de toutes les empruntes du projet. Recherche du typon sur OrCad															
Commande des composants Sous-traitance du Circuit imprimé															
Étude des algorithmes des fonctions logicielles en commençant par les plus difficiles. Test de ces fonctions sur carte MiniPicRs232.															
Fonction par fonction, soudage des composants et validation par test.															
Implantation du microcontrôleur. Ajout des fonctions logicielles et essais au fur et à mesure.															
Relever des mesures prouvant que le cahier des charges est respecté. Montage de la carte dans le boîtier prévu															
Élaboration du compte rendu															

Mesures de validation :

Pour l'instant tout fonctionne.

Fonctions logicielles :

Organisation en mémoire EEPROM Externe :

Auparavant il y a une entête de 10 octets pour connaître les caractéristiques de l'enregistrement.

- **1er octet** = N° du fichier de la mesure de 0 à 255.
- **2ème et 3ème octets** = Nombre de mesures de 0 à 65535; U et I font une mesure.
- **4ème octet** = temps entre 2 mesures de 0 à 255 s.
- **5ème octet** = Calibre ampèremètrique choisi de 0 à 255 mV/A.
- **6ème à 10ème octets** = réserve pour extensions futures.

Ensuite la tension et le courant sont codés sur 16 bits suivant le principe :

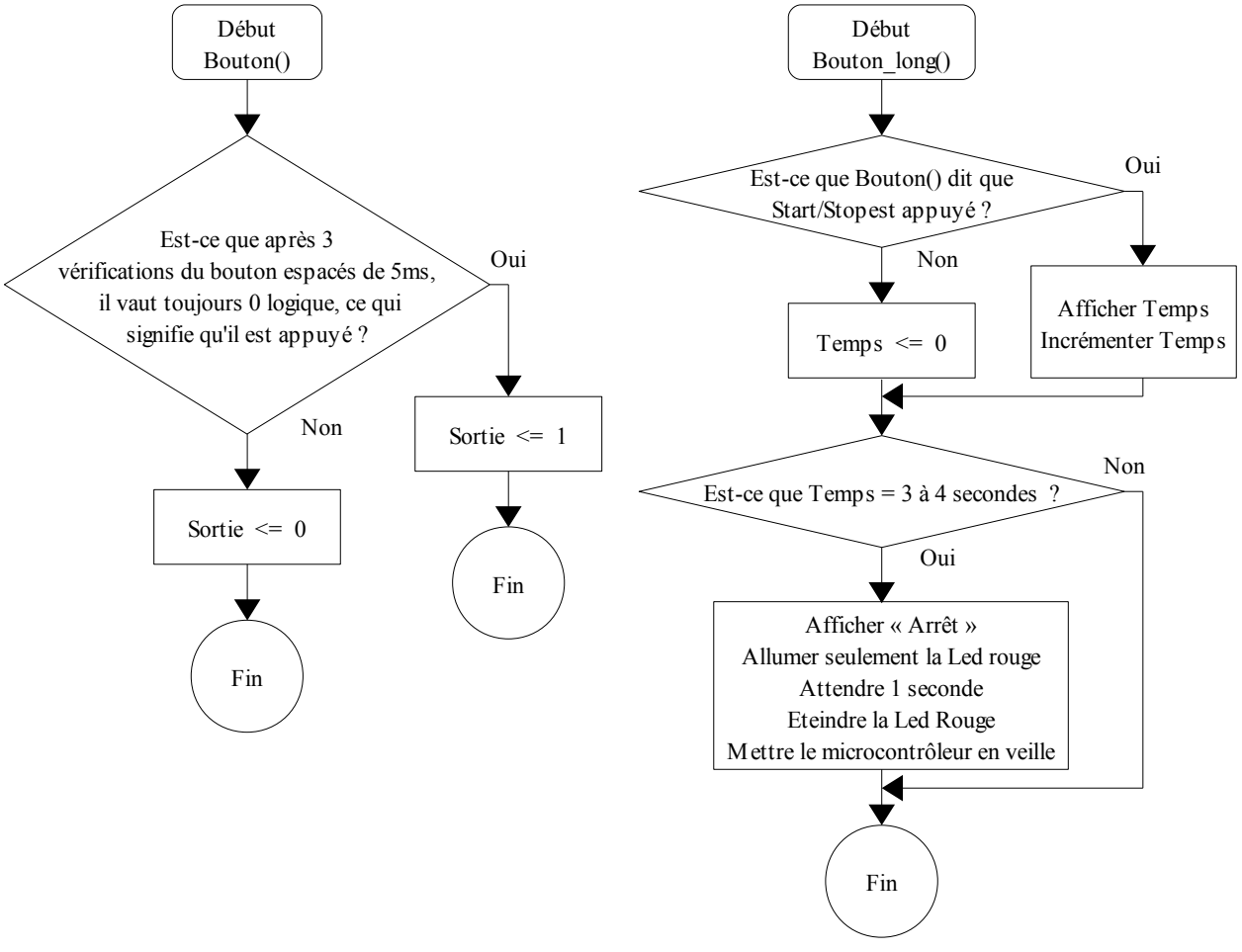
- **Tension** = -500 à 500V avec les codes -500 à 500. (possibilité de -327680 à +32767 en signed int16).
- **Courant** = -150 à +150A avec les codes -150 à +150 (possibilité de -32768 à 32767 en signed int16).

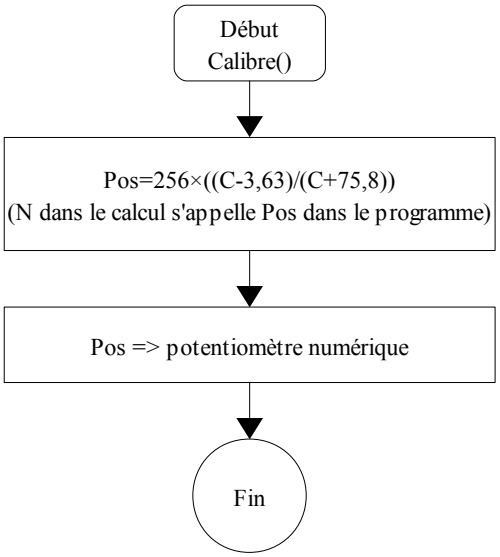
Les informations sont stockées les unes à la suite des autres, U_0, I_0, U_1, I_1, U_2, I_2, ... U_n, I_n ..... U_{fin}, I_{fin}.

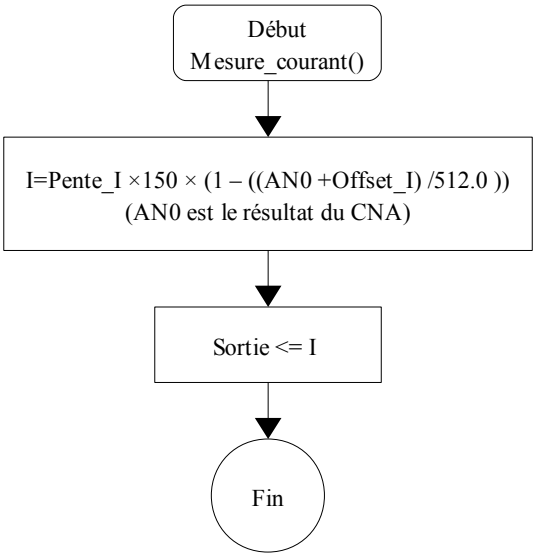
Organisation en mémoire EEPROM Interne :

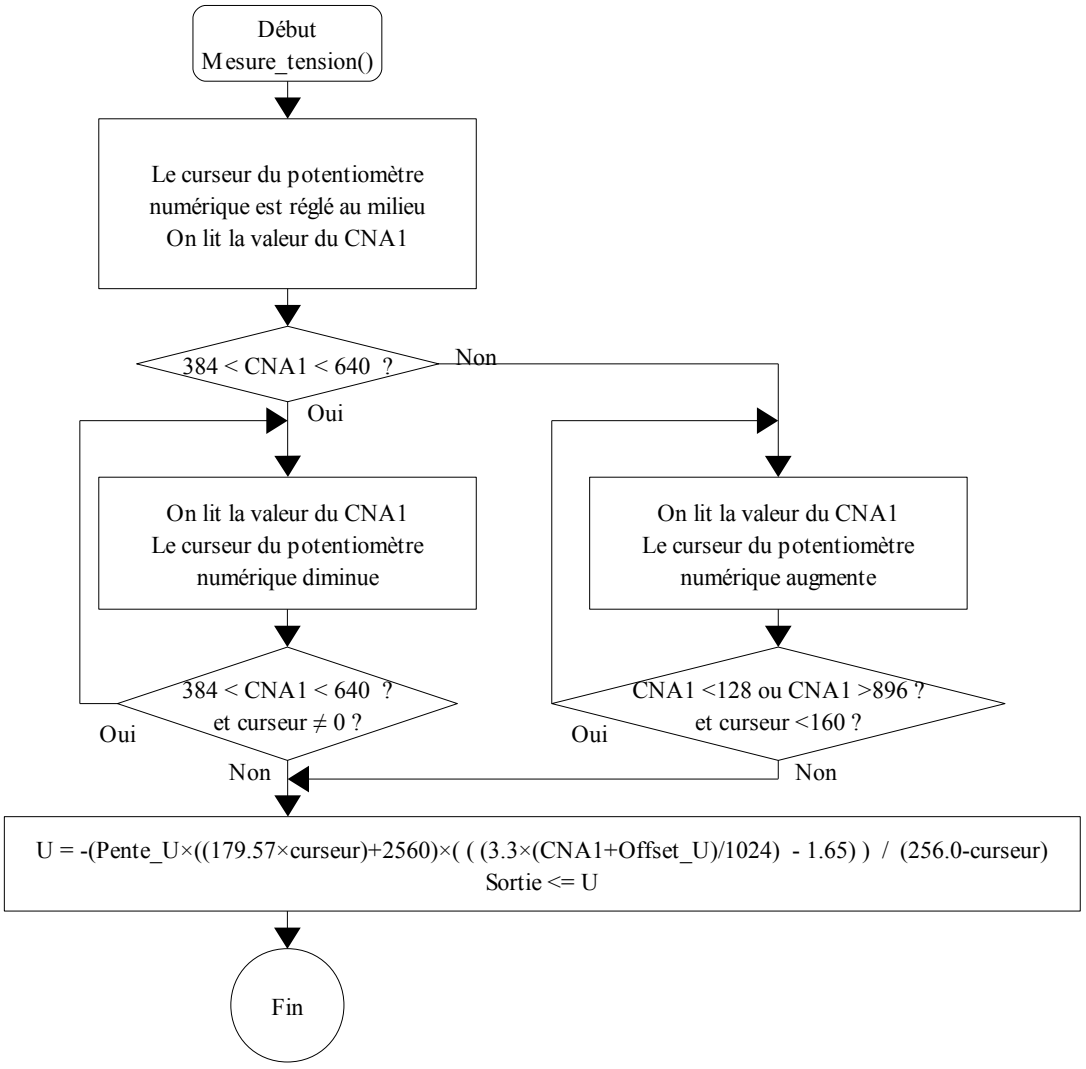
- **Adresse 10=** Année.
- **Adresse 11=** Temps d'échantillonnage sélectionné.
- **Adresse 12=** Calibre de pince ampèremètrique choisi.
- **Adresse 13=** Numéro de fichier choisi.
- **Adresse 14=** Coefficient d'étalonnage d'intensité choisi.

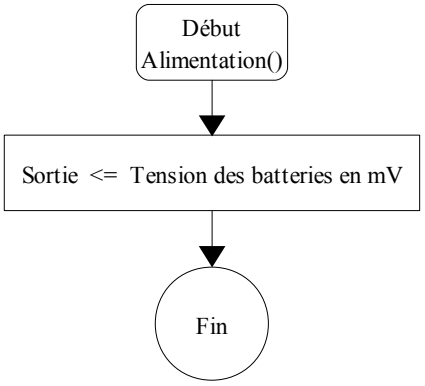
Algorithmes :

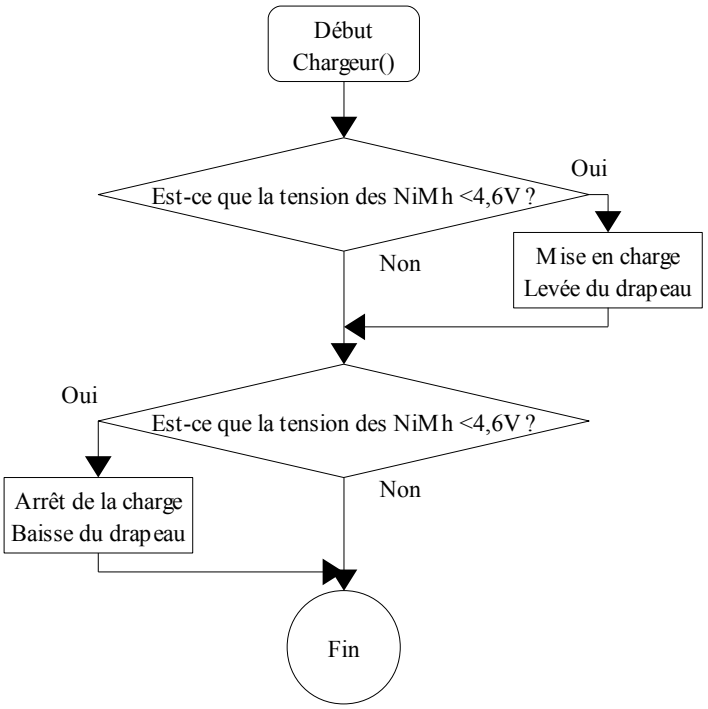



Début
Calibre()
Pos=256×((C-3,63)/(C+75,8))
(N dans le calcul s'appelle Pos dans le programme)
Pos => potentiomètre numérique
Fin


Début
Mesure_courant()
I=Pente_I ×150 × (1 – ((AN0 +Offset_I) /512.0))
(AN0 est le résultat du CNA)
Sortie <= I
Fin


Début
Mesure_tension()
Le curseur du potentiomètre numérique est réglé au milieu
On lit la valeur du CNA1
384 < CNA1 < 640 ?
Non
Oui
On lit la valeur du CNA1
Le curseur du potentiomètre numérique diminue
On lit la valeur du CNA1
Le curseur du potentiomètre numérique augmente
384 < CNA1 < 640 ?
et curseur ≠ 0 ?
Oui
Non
CNA1 <128 ou CNA1 >896 ?
et curseur <160 ?
Oui
Non
U = -(Pente_U×((179.57×curseur)+2560)×(((3.3×(CNA1+Offset_U)/1024) - 1.65)) / (256.0-curseur)
Sortie <= U
Fin


Début
Alimentation()
Sortie <= Tension des batteries en mV
Fin


Début
Chargeur()
Est-ce que la tension des NiMh <4,6V ?
Oui
Non
Mise en charge
Levée du drapeau
Est-ce que la tension des NiMh <4,6V ?
Oui
Non
Arrêt de la charge
Baisse du drapeau
Fin

Programme langage C :

```
#include <16F876A.h>
#device adc=10
#use delay(clock=20000000)
#fuses NOWDT,HS, NOPUT, PROTECT, NODEBUG, NOBROWNOUT, NOLVP, NOCPD, NOWRT
#define EEPROM_SCL   PIN_C3
#define EEPROM_SDA   PIN_C4
#use rs232(baud=9600,parity=N,xmit=PIN_C6,rcv=PIN_C7,bits=8,errors)

#include <Davlib.h >
#include <PCF8583.h>          //A0=+5V       => 0xA2 écriture  => 0XA3 lecture
#include <24LC256.h>          //A0=A1=A2=0V  => 0xA0 écriture  => 0XA1 lecture
#include <DS1803.h >          //A0=A1=A2=0V  => 0x50 écriture  => 0X51 lecture
#use i2c(Master,Fast,sda=PIN_C4,scl=PIN_C3)

#define debug(x)   printf("%s=%lu ",#x,x)  // pour voir variables internes
#define fin        printf("\n\r")

#define Rouge_1   output_high(PIN_C2)
#define Rouge_0   output_low (PIN_C2)
#define Verte_1   output_high(PIN_C1)
#define Verte_0   output_low (PIN_C1)
#define Jaune_1   output_high(PIN_C0)
#define Jaune_0   output_low (PIN_C0)
#define Marche    output_high(PIN_C5)
#define Arret     output_low (PIN_C5)
#define Charge_0  output_low (PIN_B1)
#define Charge_1  output_high(PIN_B1)

#define Start     input(PIN_B0)

int8 Annee=0;           // De 0 pour 2000 à 255 pour 2255
int8 T_ech;             // Temps échantillonnage en s de 0 à 255s (4min 15S)
int8 Cal_Pince;         // Calibre pince ampèremétrique de 0 à 255 mV/A
signed int8 Pente_I;    // Réglage de Imax
signed int8 Offset_I;   // Réglage du 0 de la mesure de l'intensité
signed int8 Pente_U;    // Réglage de Umax
signed int8 Offset_U;   // Réglage du 0 de la mesure de tension
int8 Fichier;           // N° du fichier 0 à 255
unsigned int16 Numero;  // Numéro de la mesure, à la fin = nombre de mesures
int1 Chargement;
int16 Temps;

int1 Bouton(void)
// renvoie 1 si le bouton Start/Stop est appuyé, 0 sinon.
{
  int1 data1,data2,data3;

  data1=Start;   // anti rebond
  delay_ms(5);   // et anti parasite
  data2=Start;
  delay_ms(5);
  data3=Start;
  if((data1==0)&&(data2==0)&&(data3==0)) return(1);
  return(0);
}

void Bouton_long(void)
// Vérifie s'il y a un appui long sur le bouton Star/Stop
{
  if(Bouton()==1) {debug(Temps); Temps ++;}
  else {Temps=0; return;}
  if(Temps==30)          // entre 3 et 4 secondes
  {
```

```
    printf("Arrêt\n\r");
    Rouge_1; Verte_0; Jaune_0; delay_ms(1000); Rouge_0;
    Arret;
    Sleep();
  }
}

void Calibre_I(void)
/* Règle le gain de l'AOP numérique en fonction du calibre de la pince
ampèremétrique.      Pos = position du curseur du potentiomètre numérique.    */
{
  float Pos;

  Pos=256.0 * ((float)Cal_Pince-3.63) / ((float)Cal_Pince+75.8);
  DS1803_pot0((int8)Pos);
}

signed int16 Mesure_courant(void)
/* Récupère la tension de la pince ampèremètrique et calcule l'intensité en
fonction du calibre (de -150 à 150 A).    Entrée PIC: AN0
Entrée : aucune.
Sortie : Intensité en ampère. */
{
  float I;

  set_adc_channel(0);
  delay_us(10);
  I=((float)Pente_I+1000.0)/1000.0;
  I = I * 150.0 * (1.0 - (((float)read_adc() +(float)Offset_I) /512.0 ));
  return((signed int16)I);
}

signed int16 Mesure_tension(void)
/* Récupère la tension d'entrée et choisi le bon calibre pour une meilleure
précision puis calcule la tension réelle (de -500 à 500V).    Entrée PIC: AN1
Entrée : aucune.
Sortie : Tension de batterie en volt.  */
{
  float U;
  int16 Nb;
  int8 N=127;

  DS1803_pot1(255-N);          // Les bornes sont inversées sur le schéma
  set_adc_channel(1);
  delay_us(10);
  Nb=read_adc();
  if((384<NB)&&(NB<640))
  {
    do
    {
      Nb=read_adc();
      N--;                  // on amplifie d'avantage
      DS1803_pot1(255-N);    // Les bornes sont inversées sur le schéma
    }while((384<NB)&&(NB<640)&&(N!=0));
  }
  else
  {
    do
    {
      Nb=read_adc();
      N++;                  // on amplifi moins
      DS1803_pot1(255-N);    // Les bornes sont inversées sur le schéma
    }while((Nb<128)||(896<NB)&&(N<160));
  }
```

```
  U=((float)Pente_U+1000.0)/1000.0;
  U = -(U*((179.57*(float)N)+2560.0)*(((3.3*((float)Nb)+(float)Offset_U)/1024.0)-1.65))/(256.0-(float)N);
  return((signed int16)U);
}

int16 Alimentation(void)
/* Mesure la tension d'alimentation pour contrôler la tension des éléments NiMh.
Entrée : aucune.
Sortie : Tension NiMh en mV. Exemple 5040 veut dire 5,04V */
{
  int16 data;

  set_adc_channel(4);
  delay_us(10);
  data=read_adc();
  data=(data*58)/9;
  return(data);
}

void Chargeur(void)
{
  if(Alimentation()<4600) {Charge_1; Chargement=1;}
  if(Alimentation()>5500) {Charge_0; Chargement=0;}
}

void Transmet(void)
/* Gère la liaison avec un ordinateur à travers le bus USB ou par le BlueTooth.
M Envoi les mesures au format compatible Excel.
I Etalonne l'intensité.
U Etalonne la tension.
R Remet toutes les mesures en mémoire à 0.
C Calibre la pince ampèremétrique.
E Demande le temps d'échantillonnage.
F Demande un numéro de fichier.
H Mets l'horloge à jour.
A Affiche l'horloge.
Z Initialise le microcontrôleur sans perte des données.        */
{
  int16 U,I;
  unsigned int16 n,m;
  int8 An,Mo,Jo,He,Mi,Se=0;

  if(kbhit())
  {
    switch(getc())
    {
      case '?':   printf("M Envoie les mesures compatible Excel\n\r");
                  printf("I Etalonne l'intensité\n\r");
                  printf("U Etalonne la tension\n\r");
                  printf("R Remet toutes les mesures en mémoire à 0\n\r");
                  printf("C Calibre la pince ampèremétrique.\n\r");
                  printf("E Demande le temps d'échantillonnage\n\r");
                  printf("F Demande un numéro de fichier\n\r");
                  printf("H Mets l'horloge à jour\n\r");
                  printf("D Affiche l'horloge\n\r");
                  printf("B Tension NiMh\n\r");
                  printf("A => Arrêt appareil de mesure\n\r");
                  printf("Z Initialisation sans perte des données\n\r");
                  break;
      case 'M':   n=M24LC256_read(1);
                  n=n<<8;
                  n=n | M24LC256_read(2);
                  An=M24LC256_read(0);
                  Mo=M24LC256_read(3);
```

```
            printf("Fichier N°%d\n\r%ld mesures\n\r",An,n);
            printf("Echantillonnage de %ds\n\r",Mo);
            printf("Rang;Tension;Intensite\n\r");
            for(m=0;m<n;m++)
            {
              U = make16(M24LC256_read(10+(m*4)),M24LC256_read(11+(m*4)));
              I = make16(M24LC256_read(12+(m*4)),M24LC256_read(13+(m*4)));
              printf("%ld;%ld;%ld\n\r",m+1,U,I);
            }
            break;
    case 'I':   printf("Pente_I %3u  Offset_I %2d\n\rPente_I ?: ",Pente_I,Offset_I);
            Pente_I=get_int();
            printf("\n\rOffset_I ?: ");
            write_eeprom (0x14,Pente_I);
            Offset_I=get_int(); fin;
            write_eeprom (0x15,Offset_I);
            break;
    case 'U':   printf("Pente_U %3d  Offset_U %2d\n\rPente_U ?:",Pente_U,Offset_U);
            Pente_U=get_int();
            printf("\n\rOffset_U ?: ");
            write_eeprom (0x16,Pente_U);
            Offset_U=get_int(); fin;
            write_eeprom (0x17,Offset_U);
            break;
    case 'A':   printf("Arrêt\n\r");
            Rouge_1; Verte_0; Jaune_0; delay_ms(1000); Rouge_0;
            Arret;
            Sleep();
            break;
    case 'R':   printf("Raz memoire\n\r");
            for(n=0;n<M24LC256_SIZE;n++)
            {
              M24LC256_write(n,0);
              printf("%4lX\r",n);  // permet d'attendre les 5ms EEPROM
            }
            printf("\n");
            break;
    case 'C' :  printf("Calibre pince (mV/A) ? ");
            Cal_Pince=get_int();
            Calibre_I();fin;
            write_eeprom (0x12,Cal_Pince);
            break;
    case 'E' :  printf("Echantillonnage (s) ? ");
            T_ech=get_int();  write_eeprom (0x11,T_ech);fin;
            break;
    case 'F' :  printf("Fichier ? ");
            Fichier=get_int();  write_eeprom (0x13,Fichier);fin;
            break;
    case 'H' :  printf("Date au format JJ/MM/AA HH:MM\n\r");
            Jo=get_dcb();printf("/"); Mo=get_dcb();printf("/");
            Annee=get_dcb();printf(" ");
            He=get_dcb();printf(":"); Mi=get_dcb();printf("\n\r");
            An=Annee%4;
            PCF8583_Ecriture_date(&An,&Jo,&Mo,&He,&Mi,&Se);
            write_eeprom (0x10,Annee);
    case 'D' :  PCF8583_Lecture_date (&An,&Jo,&Mo,&He,&Mi,&Se);
            Annee=read_eeprom (0x10);
            printf("Date : ");
            print_dcb(Jo);printf("/");
            print_dcb(Mo);printf("/");
            print_dcb(Annee);printf(" ");
            print_dcb(He);printf(":");
            print_dcb(Mi);printf(":");
            print_dcb(Se);fin;
```

```
                break;
        case 'B' :  printf("NiMh: %lumV Charge=%d \n\r",Alimentation(),Chargement);
                break;
        case 'Z' :  reset_cpu();
                break;
      }
        write_eeprom (0x10,Annee);
        write_eeprom (0x11,T_ech);
        write_eeprom (0x12,Cal_Pince);
        write_eeprom (0x13,Fichier);
        write_eeprom (0x14,Pente_I);
        write_eeprom (0x15,Offset_I);
        write_eeprom (0x16,Pente_U);
        write_eeprom (0x17,Offset_U);
   }
}

void Cree_entete(void)
/* Crée l'entête identifiant le début de trame de stockage en EEPROM externe.
ad 0 : 1er  octet = N° fichier mesure.
ad 1 : 2ème et 3ème octet = Nb de mesures, U et I font une mesure.
ad 3 : 4ème octet = temps entre 2 mesures.
ad 4 : 5ème octet = Calibre ampèremètrique choisi.
ad 5 ...
ad 9 : 6ème à 10ème octets = réserve pour extensions futures.        */
{
   M24LC256_write(0, Fichier);   delay_ms(5);
   M24LC256_write(3, T_ech);     delay_ms(5);
   M24LC256_write(4, Cal_Pince); delay_ms(5);
}

void Complete_entete(void)
/* Charge dans l'entête les données connues en fin de mesure   */
{
   Numero--;
   M24LC256_write(1, make8(Numero,1)); delay_ms(5);
   M24LC256_write(2, make8(Numero,0)); delay_ms(5);
}

void Stocke_en_EEPROM(int16 Numero,unsigned int16 U, signed int16 I)
/* Stocke la mesure effectuée en EEPROM externe aux bonnes adresses. */
{
   M24LC256_write(Numero*(int16)4 + (int16)10, make8(U,1)); delay_ms(5);
   M24LC256_write(Numero*(int16)4 + (int16)11, make8(U,0)); delay_ms(5);
   M24LC256_write(Numero*(int16)4 + (int16)12, make8(I,1)); delay_ms(5);
   M24LC256_write(Numero*(int16)4 + (int16)13, make8(I,0)); delay_ms(5);
}

int8 Attend(int8 data)
/* Attend data secondes et surveille l'appui sur le bouton Start/Stop
Entrée: Temps d'attente en secondes.
Sortie: état du bouton Star/Stop */
{
   int8 Bpss=0,i;

   if(data==0) return;
   do
   {
      for(i=0;i<10;i++)
      {
         delay_ms(90);
         if(Bpss==0) Bpss=Bouton();
      }
      data--;
```

```
  }while(data!=0);
  return(Bpss);
}

void main()
{
  signed int16 U,I,adresse;
  int1 R,V;
  int8 An,Mo,Jo,He,Mi,Se;

  Marche;
  setup_adc_ports(AN0_AN1_AN2_AN4_VSS_VREF);
  setup_adc(ADC_CLOCK_INTERNAL);
  setup_timer_0(RTCC_INTERNAL|RTCC_DIV_1);
  setup_timer_1(T1_DISABLED);
  setup_timer_2(T2_DISABLED,0,1);
  setup_comparator(NC_NC_NC_NC);
  setup_vref(FALSE);
  port_b_pullups(TRUE);

  PCF8583_Init();
  M24LC256_init();
  DS1803_Init();

  Annee    =read_eeprom (0x10);
  T_ech    =read_eeprom (0x11);
  Cal_Pince=read_eeprom (0x12);
  Fichier  =read_eeprom (0x13);
  Pente_I  =read_eeprom (0x14);
  Offset_I =read_eeprom (0x15);
  Pente_U  =read_eeprom (0x16);
  Offset_U =read_eeprom (0x17);
  if((Fichier==0xFF) && (Cal_Pince==0xFF) && (T_ech==0xFF))
  {
    Fichier=1;
    Cal_Pince=10;
    T_ech=1;
    Pente_U=1;
    Offset_U=0;
    Pente_I=1;
    Offset_I=0;
    Annee=8;
  }

  PCF8583_Lecture_date (&An,&Jo,&Mo,&He,&Mi,&Se);
  printf("\n\rDate : ");
  print_dcb(Jo);printf("/");
  print_dcb(Mo);printf("/");
  print_dcb(Annee);printf(" ");
  print_dcb(He);printf(":");
  print_dcb(Mi);printf(":");
  print_dcb(Se);fin;

  printf("Echantillonnage: %us\n\r",T_ech);
  printf("Calibre Pince: %umV/A\n\r",Cal_Pince);
  printf("Numero de fichier: %u\n\r",Fichier);
  printf("Pente_U: %d\n\r",Pente_U);
  printf("Offset_U: %d\n\r",Offset_U);
  printf("Pente_I: %u\n\r",Pente_I);
  printf("Offset_I: %d\n\r",Offset_I);

  Calibre_I();

  Verte_1;
```

```
  Charge_1;
  Chargement=1;
  Temps=0;
  delay_ms(2000);
  Verte_0;
  R=0;  V=0;
  do
  {
    Transmet();
    Chargeur();

    U=Mesure_tension(); I=Mesure_Courant();
    printf("U=%3ld   I=%3ld     \r",U,I);
    if((U<-1 || 1<U) && R==0)
    {               // entrée de tension branchée ?
      Rouge_1;
      R=1;
    }
    if(U==0)
    {               // on a débranché l'entrée de tension ?
      Rouge_0;
      R=0;
    }
    if((I<-1 || 1<I) && V==0)
    {               // entrée de courant branchée ?
      Jaune_1;
      V=1;
    }
    if(I==0)
    {               // on a débranché l'entrée de courant ?
      Jaune_0;
      V=0;
    }
    Bouton_long();
  }while( R==0 || V==0 || Bouton()==0);
  Jaune_0; Rouge_0; Verte_1;
  while(Bouton()==1) {Bouton_long();}     // Attend qu'on relache le bouton.
  Verte_0;
  Numero=1;
  Cree_entete();
  do
  {
    Transmet();
    Chargeur();
    U=Mesure_tension(); I=Mesure_Courant();
    printf("Mesure: %5lu   U=%3ld   I=%3ld     \r\n",Numero,U,I);
    Stocke_en_EEPROM(Numero,U,I);
    Numero ++;          Adresse = Numero*(int16)4+(int16)10;
    V=~V;  V ? Verte_1 : verte_0;             // Clignotement Led verte
    Bouton_long();
  }while(Attend(T_ech)==0 || Adresse<M24LC256_SIZE);
  Verte_0; Jaune_1;
  while(Bouton()==1) {Bouton_long();}
  Jaune_0;
  Complete_entete();

  while(true)
  {
    Transmet();   //Attend un "Reset Z" de Transmet pour recommencer
    Chargeur();
    Bouton_long();
  }
}
```

Annexes :

Microcontrôleur PIC16F876A:

FIGURE 17-1: PIC16F87XA VOLTAGE-FREQUENCY GRAPH (INDUSTRIAL, EXTENDED)

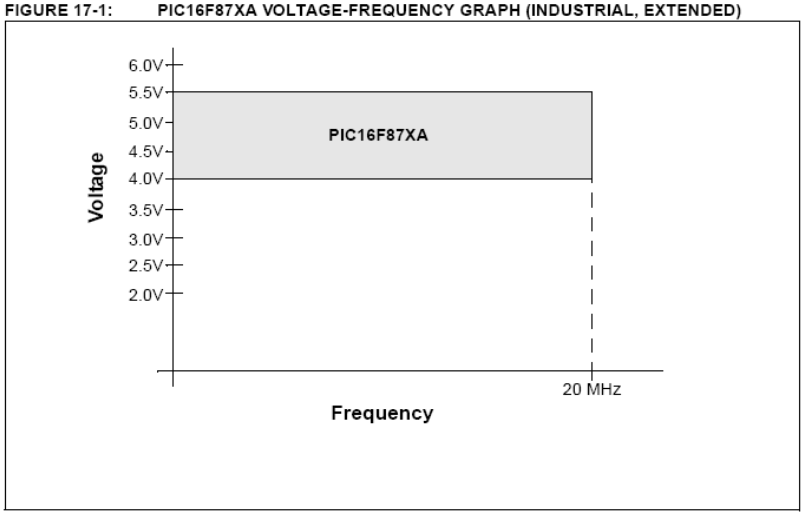


TABLE 17-14: A/D CONVERTER CHARACTERISTICS:PIC16F873A/874A/876A/877A (INDUSTRIAL)
PIC16LF873A/874A/876A/877A (INDUSTRIAL)

Param No.	Sym	Characteristic		Min	Typ†	Max	Units	Conditions
A01	NR	Resolution		—	—	10-bits	bit	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A03	EIL	Integral Linearity Error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A04	EDL	Differential Linearity Error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A06	EOFF	Offset Error		—	—	< ± 2	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A07	EGN	Gain Error		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A10	—	Monotonicity		—	guaranteed(3)	—	—	VSS ≤ VAIN ≤ VREF
A20	VREF	Reference Voltage (VREF+ – VREF-)		2.0	—	VDD + 0.3	V	
A21	VREF+	Reference Voltage High		AVDD – 2.5V		AVDD + 0.3V	V	
A22	VREF-	Reference Voltage Low		AVSS – 0.3V		VREF+ – 2.0V	V	
A25	VAIN	Analog Input Voltage		VSS – 0.3V	—	VREF + 0.3V	V	
A30	ZAIN	Recommended Impedance of Analog Voltage Source		—	—	2.5	kΩ	(Note 4)
A40	IAD	A/D Conversion Current (VDD)	PIC16F87XA	—	220	—	µA	Average current consumption when A/D is on (Note 1)
			PIC16LF87XA	—	90	—	µA	
A50	IREF	VREF Input Current (Note 2)		—	—	5	µA	During VAIN acquisition. Based on differential of VHOLD to VAIN to charge CHOLD, see Section 11.1 "A/D Acquisition Requirements".
				—	—	150	µA	During A/D conversion cycle

* These parameters are characterized but not tested.

† Data in "Typ" column is at 5V, 25°C unless otherwise stated. These parameters are for design guidance only and are not tested.

Note 1: When A/D is off, it will not consume any current other than minor leakage current. The power-down current spec includes any such leakage from the A/D module.

2: VREF current is from RA3 pin or VDD pin, whichever is selected as reference input.

3: The A/D conversion result never decreases with an increase in the input voltage and has no missing codes.

4: Maximum allowed impedance for analog voltage source is 10 kΩ. This requires higher acquisition time.

Possibilités offertes sur le Convertisseur Analogique Digital du PIC16F876A

PCFG3: PCFG0	AN7(1) RE2	AN6(1) RE1	AN5(1) RE0	AN4 RA5	AN3 RA3	AN2 RA2	AN1 RA1	AN0 RA0	VREF+	VREF-	CHAN / Refs(2)
0000	A	A	A	A	A	A	A	A	VDD	VSS	8/0
0001	A	A	A	A	VREF+	A	A	A	RA3	VSS	7/1
0010	D	D	D	A	A	A	A	A	VDD	VSS	5/0
0011	D	D	D	A	VREF+	A	A	A	RA3	VSS	4/1
0100	D	D	D	D	A	D	A	A	VDD	VSS	3/0
0101	D	D	D	D	VREF+	D	A	A	RA3	VSS	2/1
011x	D	D	D	D	D	D	D	D	VDD	VSS	0/0
1000	A	A	A	A	VREF+	VREF-	A	A	RA3	RA2	6/2
1001	D	D	A	A	A	A	A	A	VDD	VSS	6/0
1010	D	D	A	A	VREF+	A	A	A	RA3	VSS	5/1
1011	D	D	A	A	VREF+	VREF-	A	A	RA3	RA2	4/2
1100	D	D	D	A	VREF+	VREF-	A	A	RA3	RA2	3/2
1101	D	D	D	D	VREF+	VREF-	A	A	RA3	RA2	2/2
1110	D	D	D	D	D	D	D	A	VDD	VSS	1/0
1111	D	D	D	D	VREF+	VREF-	D	A	RA3	RA2	1/2

A = Analog input

D = Digital I/O

Note 1: These channels are not available on the 28-pin devices.

2: This column indicates the number of analog channels available as A/D inputs and the numer of analog channels used as voltage reference inputs.

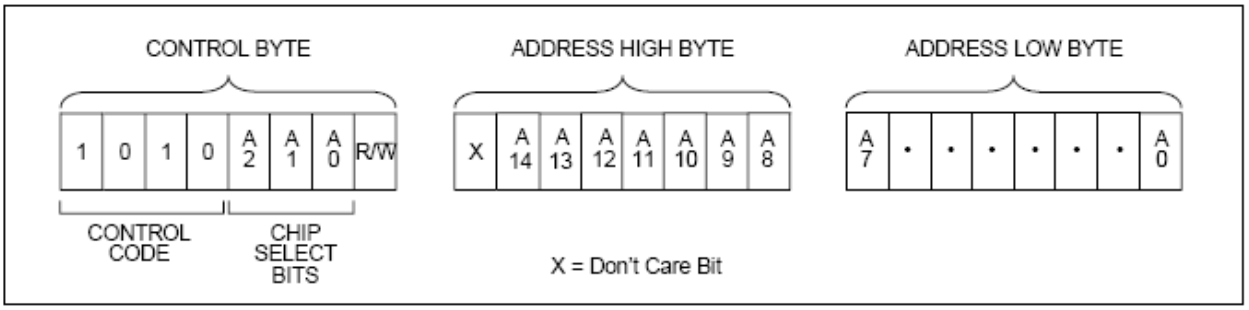
EEPROM externe 24LC256 :

DEVICE SELECTION TABLE

Part Number	Vcc Range	Max Clock Frequency	Temp Ranges
24AA256	1.8-5.5V	400 kHz†	I
24LC256	2.5-5.5V	400 kHz‡	I, E

† 100 kHz for Vcc < 2.5V.
‡ 100 kHz for E temperature range.

FIGURE 5-2: ADDRESS SEQUENCE BIT ASSIGNMENTS


CONTROL BYTE
ADDRESS HIGH BYTE
ADDRESS LOW BYTE
1 0 1 0 A2 A1 A0 R/W
X A14 A13 A12 A11 A10 A9 A8
A7 • • • • • • A0
CONTROL CODE
CHIP SELECT BITS
X = Don't Care Bit

AOP LM324 :

Characteristics	Symbol	LM224			LM324A			LM324			LM2902			LM2902V/NCV2902			Unit
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Output Voltage– High Limit ($T_A = T_{high}$ to T_{low}) (Note 5)	V_{OH}																V
V_{CC} = 5.0 V, R_L = 2.0 kΩ, T_A = 25°C		3.3	3.5	–	3.3	3.5	–	3.3	3.5	–	3.3	3.5	–	3.3	3.5	–	
V_{CC} = 30 V (26 V for LM2902, V), R_L = 2.0 kΩ		26	–	–	26	–	–	26	–	–	22	–	–	22	–	–	
V_{CC} = 30 V (26 V for LM2902, V), R_L = 10 kΩ		27	28	–	27	28	–	27	28	–	23	24	–	23	24	–	
Output Voltage – Low Limit, V_{CC} = 5.0 V, R_L = 10 kΩ, $T_A = T_{high}$ to T_{low} (Note 5)	V_{OL}	–	5.0	20	–	5.0	20	–	5.0	20	–	5.0	100	–	5.0	100	mV

MAXIMUM RATINGS (T_A = +25°C, unless otherwise noted.)

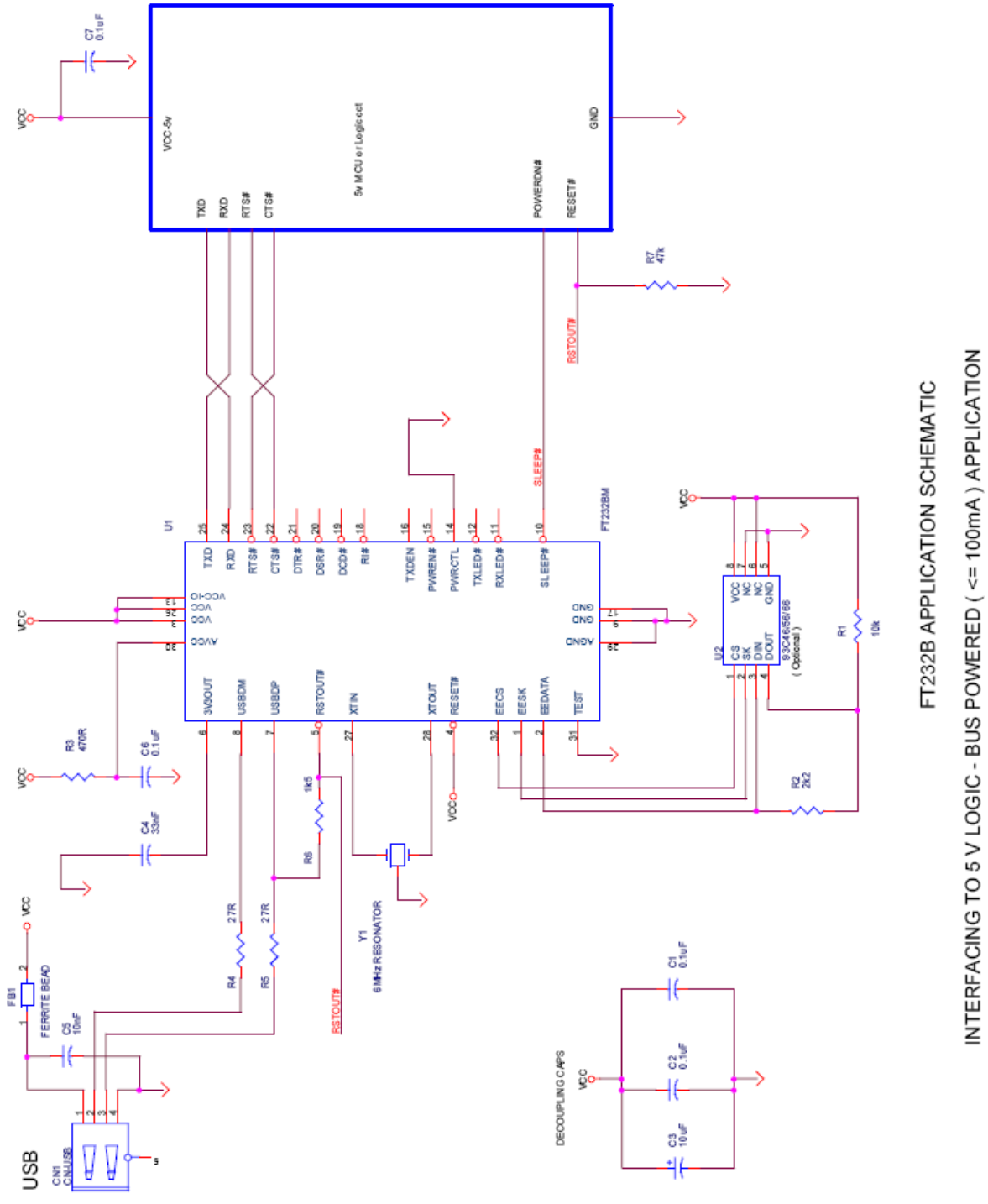
Rating	Symbol	LM224 LM324, LM324A	LM2902, LM2902V	Unit
Power Supply Voltages Single Supply Split Supplies	 V_{CC} V_{CC}, V_{EE}	 32 ±16	 26 ±13	Vdc
Input Differential Voltage Range (Note 1)	V_{IDR}	±32	±26	Vdc
Input Common Mode Voltage Range	V_{ICR}	–0.3 to 32	–0.3 to 26	Vdc
Output Short Circuit Duration	t_{SC}	Continuous		
Junction Temperature	T_J	150		°C
Storage Temperature Range	T_{stg}	–65 to +150		°C
Operating Ambient Temperature Range	T_A			°C
LM224		–25 to +85		
LM324, 324A		0 to +70		
LM2902			–40 to +105	
LM2902V, NCV2902			–40 to +125	

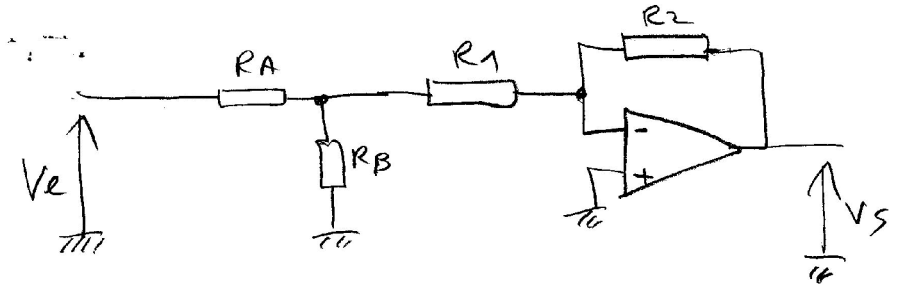
1. Split Power Supplies.

TL431 :

FT232BM:

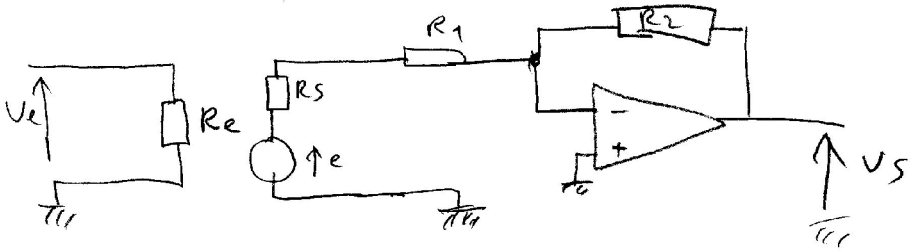
Figure 1.0 FT232BM – 5 volt Bus Powered Example Schematic (232-5VB)



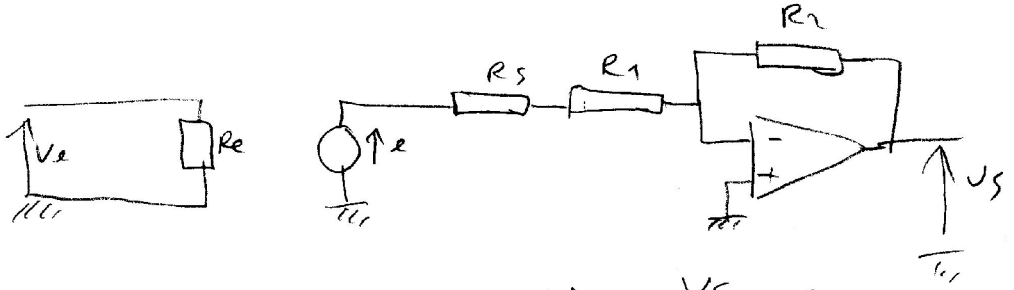


$$G = \frac{V_s}{V_e}$$

≡



≡



$e = f(V_e)$

$R_S = f(R_A, R_B)$

$$G = \frac{V_s}{V_e} = \frac{V_s}{e} \times \frac{e}{V_e}$$

$$R_S = R_A // R_B = \frac{R_A R_B}{R_A + R_B}$$

$$e = V_e \times \frac{R_B}{R_A + R_B} \longrightarrow \frac{e}{V_e} = \frac{R_B}{R_A + R_B}$$

$$G = -\frac{R_2}{R_1 + R_S} \times \frac{R_B}{R_A + R_B} = -\frac{R_2 R_B}{(R_1 + R_S)(R_A + R_B)}$$

$$G = -\frac{R_2 R_B}{\left(R_1 + \frac{R_A R_B}{R_A + R_B}\right)(R_A + R_B)} = -\frac{R_2 R_B}{\left(\frac{R_1(R_A + R_B) + R_A R_B}{R_A + R_B}\right)(R_A + R_B)}$$

$$= -\frac{R_2 R_B}{(R_1(R_A + R_B)) + R_A R_B} = -\frac{R_2 R_B}{R_1 R_A + R_1 R_B + R_A R_B}$$

$$G = -\frac{R_2}{R_1\frac{R_A}{R_B} + R_1 + R_A}$$

$$\boxed{G = -\frac{R_2}{R_1\left(1+\frac{R_A}{R_B}\right) + R_A}}$$

$R_A = 33\,k$

$R_B = 10\,k$

$R_1 + R_2 = 100\,k$

$R_1 = 0 \Rightarrow R_2 = 100 \Rightarrow G = -\dfrac{100}{0\left(1+\frac{33}{10}\right)+33} = \boxed{-3}$

$R_1 = 100 \Rightarrow R_2 = 0 \Rightarrow G = -\dfrac{0}{\ldots} = \boxed{0}$

$R_A = 33\,k$

$R_B = 1\,k$

$R_1 + R_2 = 100\,k$

$R_1 = 0 \Rightarrow R_2 = 100 \Rightarrow G = -\dfrac{100}{0\left(1+\frac{33}{1}\right)+33} = \boxed{-3}$

← donc pas d'influence sur le GAIN.

$R_1 = 100 \Rightarrow R_2 = 0 \Rightarrow G = -\dfrac{0}{\ldots} = \boxed{0}$

$R_A = 1000\,k$

$R_B = 10\,k$

$R_1 + R_2 = 100\,K$

$R_1 = 0 \Rightarrow R_2 = 100 \Rightarrow G = -\dfrac{100}{0\left(1+\frac{1000}{10}\right)+1000} = \boxed{-0,1}$

$R_1 = 100 \Rightarrow R_2 = 0 \Rightarrow G = -\dfrac{0}{\ldots} = \boxed{0}$

$$G = -\frac{R_2}{R_1\left(1+\frac{RA}{RB}\right)+RA} \qquad R_1 = \frac{PN}{255} = Pk \quad k = \frac{N}{255}$$

$$R_2 = P - \frac{PN}{255} = P - Pk$$

$$G = \frac{v}{Ic}$$

RA = 33 k
RB = ~~10~~ 5,6 k
P = 100 k
I = 150 A
v = - 1,64 ~~m~~ - 1,66 V
C = 10 ou 100 mV/A

$$G = -\frac{P-Pk}{Pk\left(1+\frac{RA}{RB}\right)+RA}$$

$$GPk\left(1+\frac{RA}{RB}\right) + GRA = -(P-Pk) = Pk - P$$

$$GP\left(1+\frac{RA}{RB}\right)k - Pk = -GRA - P$$

$$k\left(GP\left(1+\frac{RA}{RB}\right)-P\right) = -(GRA+P)$$

$$\frac{N}{255} = k = -\frac{GRA+P}{GP\left(1+\frac{RA}{RB}\right)-P} = \frac{GRA+P}{P-GP\left(1+\frac{RA}{RB}\right)}$$

$$N = 255\,\frac{GRA+P}{P-GP\left(1+\frac{RA}{RB}\right)} \qquad G = \frac{v}{Ic}$$

$$N = 255\,\frac{\frac{v}{Ic}RA + P}{P-\frac{v}{Ic}P\left(1+\frac{RA}{RB}\right)} = 255\,\frac{\frac{v}{Ic}\times\frac{RA}{P}+1}{1-\frac{v}{Ic}\left(1+\frac{RA}{RB}\right)}$$

$$N = 255\,\frac{1+\frac{v}{Ic}\times\frac{RA}{P}}{1-\frac{v}{Ic}\times\left(1+\frac{RA}{RB}\right)}$$